



**University of  
Zurich**<sup>UZH</sup>

**Department of Informatics**

# **EmbedIT – an Open Robotic Kit for Education**

A dissertation submitted to the Faculty of Economics, Business  
Administration and Information Technology of the University of Zurich

for the degree of  
Doctor of Science (Ph.D.)

by

**Dorit Assaf**

from Switzerland

Accepted on the recommendation of

Prof. Dr. Rolf Pfeifer  
Prof. Kasper Støy, Ph.D.

2013

The Faculty of Economics, Business Administration and Information Technology of the University of Zurich herewith permits the publication of the aforementioned dissertation without expressing any opinion on the views contained therein.

Zurich, January 23, 2013

Head of the Ph.D. program in informatics: Prof. Dr. Abraham Bernstein



---

# Acknowledgements

First of all, I would like to thank Prof. Dr. Rolf Pfeifer for giving me the opportunity to work in this interesting project. I highly appreciate his support at all times and his encouragement when finding funding was initially difficult. Further, I would like to thank him for providing this excellent, inspiring, crazy, international research environment. Many thanks also to Prof. Dr. Kasper Støy for being co-reviewer and to Prof. Dr. Elaine Huang for joining the Ph.D. committee. In the following lines I gratefully acknowledge many of those who directly or indirectly contributed to the successful completion of this work. Among them Dr. Beate Kuhnt who gave me the opportunity to lecture the robotics module in the context of the EFI study program. Besides the valuable teaching experience I could establish important contacts with secondary school teachers. A very special thanks to the following dedicated and motivated teachers that provided their class time and/or were available for interviews: Bettina Bieri, Bernhard Britsch, Christian Fitze, Michael Liebich, Urs Meier, Gabriel Mota, Christophe Sahli, Désirée Sterchi, Heinz Stucki, Wolfgang Pils, Lars Wieders. My gratitude further goes to Florian Wille, Moritz Kemper, and Francesco Mondada for providing me workshop time to test the toolkit. Without all of them this research would not have been possible. I cannot thank enough my good friend and colleague Lijin Aryananda who supported me in any respect, by co-lecturing, giving workshops, with technical problems, and scientific questions. Further credits go to my collaborators Fumiya Iida, Jørgen Christian Larsen, Markus Reichardt, Matěj Hoffmann, and Xiaoxiang Yu. Very special thanks to Mathias Weyland for solving my hopeless technical problems in no time and for his exquisite soldering skills, to my office mate Tao Li, and to Thomas Netter for his effort of proofreading. My sincere appreciation goes to all lab members who made this place such a wonderful working environment. Especially our lunch group which managed to cover any possible and impossible conversation topic. My appreciation also goes to our helpful current and former administrative staff: Tammar Tolcachier, Barbara Jost, Claudia Leibundgut, Sladjana Ravlija, and Claudia Wirth. My sincere thanks goes to Stephan Tschanz for his long-term support, technical advises, and his faith in me. Last but not least, I would like to thank my parents for general education, their support and encouragement.



---

# Abstract

This thesis summarizes our contribution to science and technology education by developing a novel robotic toolkit (EmbedIT). Our aim was to enable students to explore in an easy and fun way interdisciplinary topics such as biologically inspired robotics, embodied artificial intelligence, electronics, computer science, neuroscience, psychology, arts etc. The need for a new toolkit emerged based on our experiences in teaching robotics to students with different backgrounds. The current commercially available robotic kits did not fully meet our needs. Therefore, we proposed a new design approach which includes modularity, compatibility with other toolkits, compliance with industrial standards, distributed control, versatility, and open-source licensing. We emphasized a practical approach closely related to the target audience: the secondary school students. We designed a number of example exercises using EmbedIT which train a variety of skills that young people need for a successful professional life. These exercises were conducted in real class environments with a large number of students. We addressed a number of research questions on the conceptual, technical, and practical level in the context of this thesis. Pedagogues, engineers as well as teachers benefit from the insights gained, the tools developed, and the example exercises designed. This work further contributes to educational robotics research and to the dissemination of biologically inspired robotics and embodied artificial intelligence. We are sharing EmbedIT with members of the open-source community who may customize the platform to their own needs and contribute to exploit the toolkit's full potential.



---

# Zusammenfassung

Im Rahmen dieser Doktorarbeit wurde ein neuartiger Roboterbausatz (EmbedIT) entwickelt, der als Lehrmaterial den Unterricht technischer und wissenschaftlicher Schulfächer unterstützen soll. Das Ziel ist es, mit Hilfe dieses Bausatzes verschiedene interdisziplinäre Gebiete auf spielerische und einfache Art zu entdecken, zum Beispiel biologisch inspirierte Robotik, künstliche Intelligenz, Elektronik, Informatik, Neurowissenschaften, Psychologie, Kunst. Das Bedürfnis, einen neuen Bausatz zu entwickeln, entstand aufgrund unserer Erfahrungen als Lehrkräfte im Robotikunterricht mit Studenten aus verschiedenen Studienrichtungen. Die herkömmlichen, auf dem Markt erhältlichen Roboterbausätze wurden unseren Ansprüchen nicht gerecht. Wir entschieden uns für ein Design, welches die Eigenschaften Modularität, Kompatibilität mit anderen Bausätzen, Einhaltung von Industriestandards, dezentrale Steuerung, Vielseitigkeit und Open-Source-Lizenzierung aufweist. Ein hoher Stellenwert hatte die Praxisbezogenheit und Nähe zum Zielpublikum, den Schülern auf Gymnasialstufe. Wir entwickelten mit dem Bausatz einige Beispielübungen, mit denen Kompetenzen trainiert werden können, die junge Menschen heutzutage im Berufsleben benötigen. Die Übungen wurden im Unterricht an verschiedenen Gymnasien mit einer grossen Zahl von Schülern durchgeführt. In dieser Doktorarbeit befassen wir uns mit Forschungsfragen auf konzeptueller, technischer und praktischer Stufe. Von den Erkenntnissen, dem entwickelten Bausatz und den Beispielübungen profitieren Pädagogen, Ingenieure und Lehrer zugleich. Die Ergebnisse tragen zur Forschungsrichtung «educational robotics» bei und zur Popularisierung von biologisch inspirierter Robotik. Der entwickelte Roboterbausatz EmbedIT wird unter einer Open-Source-Lizenz veröffentlicht, um zu ermöglichen, dass andere die Plattform zu ihren eigenen Zwecken und für ihre eigenen Bedürfnisse anpassen und weiterentwickeln können.



---

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Background</b>	<b>7</b>
2.1	Embodied Artificial Intelligence . . . . .	7
2.2	Toolkits . . . . .	9
2.2.1	Educational robot kits . . . . .	9
2.2.2	Mechanical construction kits . . . . .	10
2.2.3	Embedded systems kits . . . . .	10
2.3	About the Target Audience – “The Digital Natives” . . . . .	11
2.4	Pedagogical Principles . . . . .	13
2.5	Educational Robotics . . . . .	13
2.5.1	Educational robotics initiatives . . . . .	14
2.5.2	Educational robotics at all levels of education . . . . .	16
2.5.3	Educational robotics – some critical reviews . . . . .	17
<b>3</b>	<b>Design Approach</b>	<b>21</b>
3.1	The Need to Develop a Novel Robotic Toolkit . . . . .	21
3.1.1	Experiences with teaching robotics . . . . .	21
3.1.2	The drawbacks of current educational robot kits . . . . .	23
3.2	Design Approach for a New Toolkit . . . . .	24
3.3	Summary . . . . .	25
<b>4</b>	<b>The EmbedIT Toolkit</b>	<b>27</b>
4.1	Hardware and Software . . . . .	27
4.2	Customizable by the Open-Source Community . . . . .	31
4.3	Mechanical Construction Components . . . . .	33
4.4	Summary . . . . .	34
<b>5</b>	<b>Teacher Interviews</b>	<b>37</b>
5.1	Summary . . . . .	40
<b>6</b>	<b>Classroom Practices and Evaluation of EmbedIT</b>	<b>41</b>
6.1	Exercise Design . . . . .	41
6.1.1	Focus on robot control . . . . .	42
6.1.2	Focus on robot morphology . . . . .	44
6.1.3	Focus on physical computing . . . . .	45
6.2	Classes and Workshops . . . . .	47
6.2.1	Control exercise results . . . . .	47

6.2.2	Morphology exercise results . . . . .	53
6.2.3	Physical computing exercise results . . . . .	54
6.3	Best Practices for Hands-on Classes . . . . .	60
6.4	Summary . . . . .	60
<b>7</b>	<b>Combined Survey Results</b>	<b>63</b>
7.1	Summary . . . . .	64
<b>8</b>	<b>Discussion and Conclusion</b>	<b>67</b>
<b>9</b>	<b>Future Work</b>	<b>71</b>
<b>A</b>	<b>Robotics as Part of an Informatics Degree Program for Teachers</b>	<b>79</b>
<b>B</b>	<b>Embedit – an Open Robotic Kit for Education</b>	<b>87</b>
<b>C</b>	<b>Extending Mechanical Construction Kits to Incorporate Passive and Compliant Elements for Educational Robotics</b>	<b>101</b>
<b>D</b>	<b>Teaching Soft Robotics: A Case Study in Building Elastic Creatures that Locomote</b>	<b>111</b>
<b>E</b>	<b>Curriculum Vitae</b>	<b>121</b>



## List of Figures

1.1	Three groups of experts involved in educational robotics: pedagogues, engineers, and teachers . . . . .	2
3.1	A collection of robots built by undergraduates of the University of Zurich in the context of the AI seminar . . . . .	22
3.2	A collection of robots used in education and competitions . . . . .	24
4.1	The EmbedIT hardware system . . . . .	28
4.2	A detailed description of each EmbedIT module . . . . .	29
4.3	Sensors and actuators . . . . .	30
4.4	Some EmbedIT modules . . . . .	31
4.5	The EmbedITApp . . . . .	32
4.6	The DC motor control panel . . . . .	32
4.7	The servo motor control panel . . . . .	33
4.8	The Architecture of the EmbedIT toolkit . . . . .	34
5.1	Ten teachers rating the importance of different properties of an educational toolkit . . . . .	39
6.1	Robots used in class for the control exercise . . . . .	43
6.2	The Braitenberg vehicles used in the context of the MINT workshop at FGZ . . . . .	43
6.3	The robot used for the Braitenberg vehicles exercise . . . . .	44
6.4	A cartoon of Rube Goldberg . . . . .	46
6.5	Students programming locomotion gaits on the Puppy robot in the context of an EFI class at Gymnasium Rychenberg . . . . .	49
6.6	Combined survey results of the EFI classes at Gymnasium Rychenberg and Gymnasium Immensee . . . . .	50
6.7	Combined survey results of the EFI classes at Gymnasium Rychenberg and Gymnasium Immensee concerning the estimated difficulty . . . . .	50
6.8	Combined survey results of the EFI classes at Gymnasium Rychenberg and Gymnasium Immensee concerning the actual difficulty . . . . .	51
6.9	The workshop description “pilote des robots marcheurs” at the EPFL Robotics Festival 2012 . . . . .	52
6.10	Students programming the Puppy and Strandbeest robots at the EPFL Robotic Festival 2012 . . . . .	52
6.11	The survey result of the Braitenberg vehicles class . . . . .	53
6.12	Students of Kantonsschule Trogen and Seetal are building soft-bodied robots for locomotion . . . . .	55
6.13	Combined survey results of the soft-bodied robot locomotion exercise of both Kantonsschule Trogen and Seetal . . . . .	55
6.14	A collection of soft-bodied robots built by students of the Kantonsschule Trogen and Seetal . . . . .	56
6.15	The constructed Rube Goldberg machine at ZHdK . . . . .	57
6.16	Students working on a Rube Goldberg machine at ZHdK . . . . .	58
6.17	Survey results of the Rube Goldberg machine exercise at the workshop day of ZHdK “Junior Design Department” . . . . .	58
6.18	Students of the Kantonsschule Trogen are building a Rube Goldberg machine . . . . .	59
6.19	Survey results of the Rube Goldberg exercise at the Kantonsschule Trogen . . . . .	59
6.20	Survey results of the ubiquitous computing exercise at the EFI class of Kollegium Spiritus Sanctus Brig . . . . .	60

7.1	Twelve students from Kantonsschule Trogen compared the two exercises Rube Goldberg Machine and soft-bodied locomotion . . . . .	64
7.2	Combined survey results of all data concerning the exercises . . . . .	64
7.3	Combined survey results of all data concerning the EmbedIT toolkit . . . . .	65

## List of Tables

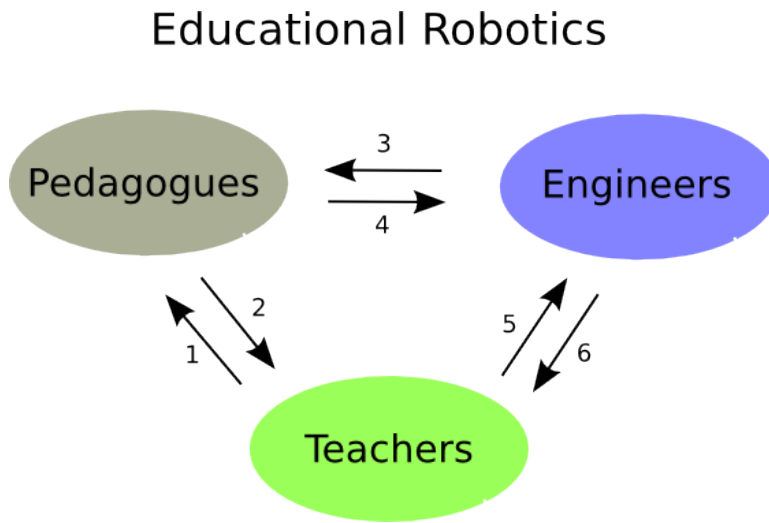
2.1	A collection of fixed educational robot platforms . . . . .	19
2.2	A collection of flexible educational robot platforms . . . . .	20
6.1	Overview of all classes and workshops conducted in the context of the EmbedIT evaluation . . . . .	48
8.1	Feature comparison between LEGO Mindstorms and EmbedIT . . . . .	69

# Introduction

*“We live in a society exquisitely dependent on science and technology, in which hardly anyone knows anything about science and technology.”* [Sagan, 1990] With this statement the American astronomer and science popularizer Carl Sagan referred in 1990 to the increasing science illiteracy among people in the United States. Also in Europe we observe the same phenomenon, namely a decreasing interest in science, technology, engineering, and mathematics. Our society has been highly influenced by emerging information and communication technologies in the last few decades. This rapid development has only started, great changes are yet to come. Mobile phones and the Internet changed the life of billions of people around the globe. These new technologies offer great opportunities. At the same time we are facing unfamiliar threats such as climate change and shortage of natural resources, which are resulting in increasing natural disasters, economic and environmental refugees as well as wars on commodities. Science and technology are the disciplines that can drive the progress in solving these issues. Consequently, education in these fields is crucial not only for humanity as a whole also for each country’s competitiveness and well-being. Or more succinctly put by Sagan: *“Public understanding of science is more central to our national security than half a dozen strategic weapons systems.”* [Sagan, 1990]

This thesis summarizes our attempt to contribute to science and technology education. We developed an open-source educational robot kit (EmbedIT) that enables easy and fun exploration of a variety of interdisciplinary fields such as (biologically inspired) robotics, embodied artificial intelligence (AI), electronics, computer science, neuroscience, psychology, arts, etc. We designed a number of novel exercises using EmbedIT which train a diversity of important skills such as logical reasoning, scientific thinking, interdisciplinary thinking, brainstorming, creative prototyping, mechanical construction, team work, time management, etc. The students are further confronted with concepts of embodied artificial intelligence such as the frame-of-reference problem (i.e. what the world “looks like” from an agent’s perspective), emergent behavior, individual vs. collective perspectives, etc. Our goal was to design a platform which enables the exploration of these concepts and the training of the above mentioned skills. Traditional educational robotic platforms often focus on a typical engineering approach, where the emphasis lies on programming a pre-defined platform. We believe that interdisciplinary skills are needed to cope with challenges an increasingly complex world brings about. We address in this thesis a number of research questions regarding conceptual design principles, technical challenges, classroom practices, and evaluation issues. The gained insights contribute to educational robotics research as a whole and to all individual expert groups involved in this interdisciplinary field (as illustrated in Figure 1.1)

The emerging research field of educational robotics addresses both educational questions and technological problems. This interdisciplinary field aims to improve science and technology education by conducting hands-on exercises often with the use of robotic platforms (see section 2.5 for more information). Ideally, three groups of experts collaborate closely together in an ed-



**Figure 1.1:** Three groups of experts involved in educational robotics: pedagogues, engineers, and teachers. Pedagogues contribute theoretical pedagogical principles (e.g. teaching and learning styles). The engineers develop toolkits and provide technical solutions. The teachers have the most direct contact to the target audience (students). They adapt the theoretical teaching styles, create and conduct exercises using the developed toolkits. Ideally, there should be a tight collaboration among all three groups (arrows 1-6) throughout an educational robotics project.

educational robotics project: pedagogues, engineers, and teachers (Figure 1.1). The pedagogues provide theoretical principles on teaching and learning methods (e.g. discovery learning, instructional teaching etc.). They are interested in how knowledge is constructed and which learning methods trigger the required cognitive processes. The engineers develop technical educational material (e.g. toolkits). They aim to find new technical solutions to support teaching and learning in classroom environments. The teachers have the closest contact to the target audience, the students. They possess great practical knowledge about all aspects of teaching. They adapt pedagogical methods, create and conduct exercises using the toolkits developed by the engineers. The teachers are key for the success of any educational robotic projects. They need to be on board and fully convinced by both the toolkit and the educational principles. Each expert group should constructively influence each other throughout the entire project (arrow 1-6 in Figure 1.1): 1. Pedagogues provide teaching and learning methods to the teachers, 2. Teachers give “real world” feedback to the pedagogues, 3. Engineers provide tools to conduct experiments, further they define which technical skills have to be learned. 4. Pedagogues give insights how a toolkit should be designed to improve learning and teaching. 5. Engineers deliver technical teaching material (toolkits) as well as technical support and training. 6. Teachers give valuable practical feedback about any aspect of the toolkit.

In this thesis, we developed a novel robotic platform that enables easy and fun exploration of a variety of interdisciplinary fields. This decision came after we have used a number of commercially available educational robot kits for our own teaching purposes. We found that even though there is a large number of products available, their variety is limited. Manufacturers as well as research laboratories tend to create their custom platforms, often with similar features but proprietary standards. The exercises usually focus on programming a predefined, wheeled robot which highly constrains the diversity of skills that can be trained. We believe that science and engineering involve interdisciplinary topics which cannot be trained with programming tasks only. Even the most flexible and prevalent platform, LEGO Mindstorms, does not fully meet our needs,

since we focus on biologically inspired robotics which includes for instance the exploration of soft material properties and passive dynamics. Further, computer science and engineering is a fairly new topic taught in secondary schools in Switzerland (this applies to many other countries as well). There are no surveys to date how teachers design the curriculum of these classes and what their personal requirements are for an educational toolkit. The field of educational robotics still lacks to date of qualitative and quantitative long-term studies about the impact of initiatives on the actual career choices of the students.

Based on the open issues in educational robotics research as well as on the drawbacks found in state-of-the-art educational robotic platforms, we address in this thesis a number of research questions (see list below). All research questions are most certainly relevant for all expert groups of Figure 1.1. Nevertheless, they can be assigned to each group as follows: The conceptual, technical, and questions about the teachers are mostly relevant for engineers. The classroom practice questions concern the teachers, and the evaluation issues are interesting for pedagogues. Furthermore, next to the individual benefits for each expert group, the insights gained in the context of this thesis contribute to the research field of educational robotics as a whole.

*Conceptual questions:*

1. How should a platform be designed to enable versatility, adaptability, and scalability?
2. How should a platform be designed to explore principles of biologically inspired robotics and embodied artificial intelligence?

*Technical challenges:*

3. How should the platform architecture and its interfaces be designed to meet the requirements of question 1–2?
4. Which are the technical challenges in developing a novel platform and what are the bottlenecks of the chosen design?

*Questions about the teachers:*

5. What are the classroom practices of teachers giving computer science and engineering classes in secondary schools?
6. What are the teachers' requirements for educational robotic toolkits?

*Classroom practice questions:*

7. Is it possible to design exercises, which train a diversity of skills using just one platform?
8. How does EmbedIT perform in real classroom environments?
9. What are the best practices to conduct hands-on classes using toolkits?

*Evaluation issues:*

10. How can educational robotics classes be evaluated?

In the context of this thesis, we created concepts, developed a toolkit and a number of novel example exercises. Further, we conducted studies about the teachers as well as collected data from real class experiments. We conclude that this work contributes to each expert group as follows:

The teachers can consult the reviews of educational robot kits and mechanical construction kits to find an adequate platform that meets their needs. We created a repository of creative exercises which could either be directly adopted by the teachers or customized according to their needs. The same exercises can also be carried out with different toolkits (if possible) or EmbedIT can be used for other exercises. The best practices are furthermore beneficial, and the teacher interviews give insights on their colleagues' work.

Our design concepts, the information about the teachers as well as the implementation of the platform contribute to the engineers. They can profit from the open-source license, which enables them to either adopt the hardware and software as is or customize it according to their needs. Our technical design approach can give important recommendations for any toolkit developer.

The pedagogues profit from the survey results as well as from the developed toolkit and the example exercises. It enables them to conduct specific studies on teaching and learning methods. The morphology exercise for instance would be a useful example to conduct studies in discovery learning. It is very easy and cheap to conduct this exercise with a large number of students.

This thesis is structured around the research questions. Each chapter summarizes the conclusions of its respective research questions. The document organization is as follows:

**Chapter 2** summarizes related work. Section 2.1 gives a brief overview on principles of embodied artificial intelligence, a research field that highly influenced our development. Section 2.2 reviews current commercially available educational robot kits. Section 2.3 elaborates the characteristics and learning preferences of the target audience—today's students. We took these principles into consideration in the design of our toolkit. This is followed by section 2.4, which addresses modern pedagogical principles. Section 2.5 completes the background information with a review on the research field of educational robotics.

**Chapter 3** addresses the conceptual questions 1–2. Our approach was to draw conclusions from our own teaching experiences using state-of-the-art robotic platforms. Further, we took current findings about the target audience as well as modern pedagogical principles into consideration.

**Chapter 4** focuses on the technical questions 3–4. We specify technical details of both hardware and software of the toolkit and outline our design approach.

**Chapter 5** covers research questions 5–6 which relate to the teachers. We describe our insights gained by conducting teacher interviews. We targeted teachers that are giving computer science and engineering classes in secondary schools.

**Chapter 6** is dedicated to research questions 7–9. We designed different example exercises that prove the versatility of the toolkit. We show solutions how to reduce preparation efforts and costs. Throughout this thesis we made a point of acquiring "real life" feedback about our concepts and prototypes. Therefore we thoroughly tested the exercises and the toolkit in a number of real class environments.

**Chapter 7** combines all survey results acquired during the test runs in schools. Here, we try to find answers for question 10.

**Chapter 8** summarizes all insights gained by addressing the research questions.

**Chapter 9** outlines the future work of the project, open issues and possible long-term vision.

---

Furthermore, this thesis consists of a number of papers both published and unpublished (to date):

**Appendix A: Published**

Assaf, D. and Pfeifer, R. (2011). *Robotics as Part of an Informatics Degree Program for Teachers*, In Proceedings of Society for Information Technology & Teacher Education International Conference 2011, pages 3128–3133. Chesapeake, VA: AACE.

**Appendix B: Published**

Assaf, D. and Pfeifer, R. (2011). *Robotics as Part of an Informatics Degree Program for Teachers*, In Research and Education in Robotics – EUROBOT 2011, Communications in Computer and Information Science, volume 161, pages 29–39. Springer.

**Appendix C: Published**

Assaf, D. and Larsen, J. C. and Reichardt, M.(2012) *Extending mechanical construction kits to incorporate passive and compliant elements for educational robotics.*, In Proceedings of 3rd International Conference on Robotics in Education (RiE 2012), pages 33–40.

**Appendix D: Submitted**

Assaf, D. and Yu, X. and Wang, L. and Iida, F. (2012) *Teaching soft-robotics: A case study in building elastic creatures that locomote*





# Background

In this chapter we summarize the related work which is relevant for this thesis. Section 2.1 provides insights about the field of “embodied artificial intelligence”, which has formed the major research target of our laboratory (AILAB) in the last 15 years. Our aim is to teach the principles of embodied AI and biologically inspired robotics by using an appropriate educational toolkit.

There are a variety of robotic toolkits available off-the-shelf. We tested several to find the one best suited to our needs. Based on our survey we decided to build our own novel educational robot kit. A review of commercially available toolkits is in section 2.2.

The AILAB has a tradition in teaching robotics and embodied AI to students with different backgrounds and from different academic levels, such as secondary schools, undergraduates and postgraduates [Paul et al., 2000]. It is important to consider the characteristics, technology knowledge as well as learning preferences of students when developing and testing the toolkit. To learn more about this target audience refer to section 2.3.

To develop a useful toolkit that meets pedagogical needs, we incorporated state-of-the-art educational principles in the design of the toolkit and the classes where it is used. A brief summary of relevant educational principles can be found in section 2.4.

As a response to decreasing enrollments in engineering-related disciplines in the last decades and the consequent lack of skilled workforce, educators from all academic levels try to find solutions to motivate students to pursue a career in these fields. Robots have increasingly been used in the classroom as a hands-on learning tool. The research field of educational robotics therefore grew considerably over the last decade. Section 2.5 gives more detail about this field and provides an overview of the respective literature.

## 2.1 Embodied Artificial Intelligence

We briefly explain in this section the main concepts of embodied artificial intelligence. We highly recommend the related literature for further reading, e.g. [Pfeifer and Scheier, 1999], [Pfeifer and Bongard, 2006], [Pfeifer et al., 2007], [Hoffmann and Pfeifer, 2012]. There is additionally an online tutorial available which includes numerous pictures, videos, links to online lectures, and exercises: *Tutorial on Embodiment*, EUCOG website<sup>1</sup>, M. Hoffmann and D. Assaf (2011).

The term embodied artificial intelligence was introduced in the mid-1980s as a response to “classical” artificial intelligence. The classical approach focuses more on algorithms and control programs, since intelligence is viewed as solely a matter of abstract symbol processing. With increasing computational power, classical AI algorithms such as in machine-learning or search

---

<sup>1</sup><http://www.eucognition.org/index.php?page=tutorial-on-embodiment>

programs became very successful. Today, they are implemented ubiquitously in applications including Internet search, automatic completion for text-processing programs, natural language interfaces (e.g. Siri on iPhone), data-mining, computer games, control for appliances, home electronics, cars, trains, etc. The classical approach, however, does not perform so well when it comes to understanding natural forms of intelligence such as recognizing a face in a crowd, manipulations of flexible, soft objects and materials, walking, running, riding a bicycle, producing and understanding natural language, etc. The notion of embodied intelligence therefore states that in order to understand intelligent behavior in humans, animals, or machines, it is necessary to consider their embodiment and embedding. That is,

*"the behavior of any system is not merely the outcome of an internal control structure (such as the central nervous system). A system's behavior is also affected by the ecological niche in which the system is physically embedded, by its morphology (the shape of its body and limbs, as well as the type and placement of sensors and effectors), and by the material properties of the elements composing the morphology."* [Pfeifer et al., 2007]

We point out two important aspects of studying intelligence: the synthetic methodology and the frame of reference problem.

#### **The synthetic approach:**

*"We distinguish between analytic and synthetic approaches. The analytic approach is universally applied in all empirical sciences. Typically, experiments are performed on an existing system, a human, a desert ant, or a brain region, and the results are analyzed in various ways. Often the goal is to develop a model to predict the outcome of future experiments. By contrast, the synthetic approach works by creating an artificial system that reproduces certain aspects of a natural system. This is another important function of models. Rather than focusing on producing the correct experimental results, that is, the correct output, we can try to reproduce the internal mechanisms that have led to the particular results. In a memory experiment, we could predict, say, the number of items recalled, based on a statistical model. Alternatively we could try to model the memory processes themselves. An ethologist may want to predict where an ant path will be formed. Again, he can use statistical modeling, but he can also attempt to model the behavioral rules by which the ants interact with the environment and with each other. Such models are typically computer models that, when run, are expected to reproduce the experimental results. The focus of interest shifts from reproducing the results of an experiment, although that is still an important aspect, to understanding why the results come about. This kind of approach is called synthetic modeling and is extremely productive. It is at the core of the discipline embodied cognitive science. Such an approach can be characterized as "understanding by building".*" [Pfeifer and Scheier, 1999], pp. 21-22.

#### **The frame-of-reference problem:**

*"Agents can exploit physical laws even if they are not aware of them. Intelligence, in this sense, is not so much a property of an agent or of the brain or of evolution, but rather resides in the eye of the beholder, so to speak, who observes the exploitation. This leads us to the next general problem that the study of intelligence raises: the so-called frame-of-reference problem. The frame-of-reference issue, which is concerned with the perspectives that we can adopt when observing or designing agents, implies that we must be very clear about what we are observing and how we interpret what we observe. The initial inspiration for this line of thought comes from Herb Simon's seminal book "The Sciences of the Artificial", in which he introduced the anecdote of an ant walking along a beach [Simon, 1976]. He argued that from an observer's point of view, the ant describes a complex path because it walks around puddles, rocks, twigs, and pebbles. However, from the point of view of the ant, the mechanisms that bring about*

*this behavior might in fact be quite simple, such as "if obstacle on right then turn left" or "if obstacle on left then turn right," and "go straight." The final path of the ant emerges from its interaction with the environment; in this case, a beach. The ant knows nothing about puddles, pebbles, and twigs but still manages to find its way around quite well (see also [Pfeifer and Scheier, 1999]).*" [Pfeifer and Bongard, 2006], pp. 72.

We aim to provide a toolkit that enables students to explore principles of embodied intelligence in a user-friendly and fun way, by using the synthetic methodology, or the "bottom-up" approach. The current state-of-the-art educational robot kits focus on a "top-down" approach, which is generally used in classical AI.

## 2.2 Toolkits

Robots consist in a body comprising sensors, actuators, batteries, control boards, communication units, and additional electronic components. All these components can either be purchased individually or composed in a toolkit. The toolkit is usually cost-saving and convenient, since its components are chosen to be compatible with each other. Building a robot from scratch from individually collected parts requires profound knowledge in mechatronics. We cannot expect this from teachers and students, consequently, using a toolkit is the right choice for education. There is a great variety of commercially available toolkits, and finding the right one for one's purpose is a time-consuming task. To break down the search, we distinguish three main categories of toolkits, all of which are useful to build robots but differ in their completeness.

The first category is the "embedded systems kits". Control boards are usually the "brain" of a robot. These printed circuit boards (PCB) equipped with a microcontroller collect sensor data and control actuators. The user implements the robot's control program on a PC using Assembler, C, or C++ programming languages and uploads the executable program to the microcontroller, which afterwards runs independently. Such kits advantageously enable the robot builder to bypass PCB design and component soldering while facilitating microcontroller software development. The embedded systems kits, which cover exclusively the control part of a robot, are described in section 2.2.3.

"Mechanical construction kits" is the second category of toolkits, which solely cover the mechanical construction part of a robot. With this term we refer to construction systems usually comprising re-usable elements such as strips, plates, angle girders, axles and gears with nuts and bolts to connect the pieces. The elements can be made out of plastic or metal, the connections can be screwed or cemented. Mechanical construction kits are the most traditional kits of the categories mentioned here, some were invented a hundred years ago. A detailed review on mechanical construction kits can be found in section 2.2.2

The third category is the "educational robot kits". Robots have been used in the last decade to introduce kids to science and technology, consequently, a large number of off-the-shelf educational robot kits exists. With robot kits we refer to robotic platforms, which provide sensors, actuators, control boards as well as mechanical construction components. This is the most complete group of toolkits, which is discussed in section 2.2.1.

### 2.2.1 Educational robot kits

We list in this section a collection of commercially available educational robot kits from all over the world. We categorized them in two groups, fixed (Table 2.1) and flexible platforms (Table 2.2). Unlike the fixed platforms, the flexible provide a set of mechanical construction components to custom build the body of a robot. The collections of Table 2.1 & 2.2 are far from complete, but the

listed robotic platforms are a good representation of state-of-the-art educational robot kits. Especially among the fixed platforms there is a great range regarding price, quality, and complexity. There is basically a platform for every need and budget. Hobbyists and teachers would go for the cheaper, less complex products (price below EUR 200), the more complex and industry related ones might be targeted for university level education (e.g. ePuck, Robotino, NAO). With exception of NAO which is the only humanoid among the fixed platforms, all the others are more or less similar, i.e. car-like robots with standard sensors to navigate autonomously. Topobo is the only platform among the flexible ones that follows a different approach (programming through kinetic memory). All the other kits are conceptually very similar. They provide standard actuators and sensors, a central control unit (with graphical or textual programming), mechanical construction parts made out of plastic or metal, screwed or snapped together.

## 2.2.2 Mechanical construction kits

A review on mechanical construction kits can be found in chapter 2B of the following publication (Appendix C) [Assaf et al., 2012a].

Dorit Assaf, Jørgen Christian Larsen, and Markus Reichardt

### *Extending Mechanical Construction Kits to Incorporate Passive and Compliant Elements for Educational Robotics*

**Abstract** Robots are a popular educational tool to introduce science, technology, and engineering to students. The field of educational robotics is growing and consequently a number of educational robot kits have been developed within the last decade. Our laboratories have a tradition of teaching embodied artificial intelligence and biomechanics to students with different backgrounds. The robots we use both for research and education are usually built incorporating compliant materials as well as passive dynamics. These kind of properties are often not available in classical robot kits or mechanical construction kits. In this paper we describe some of the robots we use for education. So far we built the robots using 3D printing technology which is convenient but too expensive for class use. Our aim is to find cheaper, commercially available solutions. After a short review on educational robot kits and mechanical construction kits we describe interface solutions between several kits. Further we show some solutions to incorporate compliant materials and passive dynamics to traditional mechanical construction kits by using cheap and widely available materials.

## 2.2.3 Embedded systems kits

Arduino<sup>2</sup> is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. The Arduino project provides further custom C/C++ libraries to program an on board microcontroller. Hardware design and software is open-source and a large community grew around the platform by contributing libraries, tutorials, hardware designs, etc. Meanwhile, several Arduino control board designs have been developed, each suited for different applications. Even though there are plenty of well documented online tutorials, skills in electronics are still needed to attach sensors and actuators to these boards (which have to be purchased separately). In addition, the Arduino boards can be damaged easily by wrongly connecting additional components. Similar to Arduino, Dwengo<sup>3</sup> aims at making microcontrollers available to a broad audience. They provide hardware components, graphical programming capabilities and simulation environments. A fairly new project related to the EmbedIT platform is .NET Gadgeteer<sup>4</sup> of

<sup>2</sup><http://www.arduino.cc>

<sup>3</sup><http://www.dwengo.org>

<sup>4</sup><http://research.microsoft.com/en-us/projects/gadgeteer/>

Microsoft research. It is a rapid prototyping platform for small electronic gadgets and embedded hardware devices. It combines object-oriented programming and solderless assembly of electronics using a kit of hardware modules. Unlike the Arduino platform .NET Gadgeteer provides all kinds of sensors and actuators that can be plugged into the main board. This prevents problems caused by wrong connections. Similarly, Phidgets<sup>5</sup> offers a variety of sensors and motor controller modules which are all connected through a USB interface to the computer. Additionally, the Phidgets libraries support the most common programming languages.

## 2.3 About the Target Audience – “The Digital Natives”

The generation born roughly between 1980 and 2000 is considered the “Net generation” [Tapscott, 1998], “millenials” [Howe and Strauss, 2000], [Howe and Strauss, 2003] or more popularly, the “digital natives” [Prensky, 2001a]. They grew up being exposed to computers, the Internet and technological gadgets such as mobile phones, digital music players and digital cameras. For that reason they are often assumed to be universally savvy with information and communication technologies. They are set apart from previous generations being ascribed distinct characteristics in terms of learning preferences. It is asserted that digital natives are proficient in multitasking, prefer experiential activities and collaborative learning environments with the use of technology. In response to these claims fundamental questions are being raised about whether education is currently equipped to meet the needs of these students. Opposed to “digital natives” Prensky has coined those born prior to 1980 “digital immigrants”, referring to the fact that many had to familiarize themselves with digital technology at a later stage in their life. Moreover, this section of the population which includes most teachers, lacks the technological fluency of the digital natives, a phenomenon often being labeled the technological gap. Prensky further claims that an insufficient use of technology in the classroom by educators is creating rejection among the students and characterizes this as “the biggest single problem facing education today”. Prensky’s viewpoints are shared and referenced by a host of later publications [Bennett et al., 2008]. Naturally, such strong claims make massive waves and an ongoing “digital natives” debate emerged among educators, social scientists and researchers. Bennett et al. [Bennett et al., 2008] critically review the evidence of this debate and point out the fact that the above described assertions are put forward with limited empirical evidence and the sense of impending crisis pervading the debate is a result of a so called “moral panic”. Recent studies about the access and use of technology among young people in post compulsory education offer a more diverse view [Kvavik, 2005], [Oliver and Gorkerke, 2007], [Kennedy et al., 2006], [McEuen, 2001]. The research indeed confirmed a pervasive use of technology and a high connectivity among students but revealed, on the other hand, that the technological fluency is solely of basic and superficial nature. The way young people use technology seems to foster mastering skills rather than understanding.

Prensky [Prensky, 2001b] even claims that young people’s brains now function differently than the ones of previous generations as a result of extensive exposure to video games. A more moderate view by Frand [Frand, 2000] identifies ten attributes reflecting values and behaviors that make up “the information age mindset”. They address how people do things and describe subliminal needs conditioned by the cyber age. From Frand’s observations we will mention here those attributes that relate directly to our context. One fundamental fact is that for students today “computers aren’t technology”. Digital immigrants or people that possess, according to Frand, an “industrial-age mindset” grew up with telephones, cars and television. Therefore, for that generation these technologies aren’t really technology, opposed to computers, the Internet and mobile phones that are considered to be such. For the young generation technology is everything that surrounds and is made possible by computers but it is not the actual computer itself. In

---

<sup>5</sup><http://www.phidgets.com/>

the context of constructivism, a philosophy on how we come to understand or know, “cognitive conflict or puzzlement is the stimulus for learning and determines the organization and nature of what is learned” [Savery and Duffy, 1996]. Therefore, a possible reason why young people are not interested in learning about technology might be because it does not puzzle them. In contrast to the older generation that considers it to be high-tech and that can’t help but wonder with awe about the new possibilities and implications.

Some of Frand’s [Frand, 2000] attributes that reflect “the information age mindset” are:

**Doing rather than knowing:** *“The view of knowledge as a product—“a body of facts accumulated by mankind” (Webster’s Dictionary)—must be reexamined. In the past, the half-life of information was measured in decades and centuries, certainly longer than most individual’s life span. College and university students could learn skills and gain knowledge that would carry them through their careers. In many disciplines, the half-life of information is measured in months and years. From this perspective, what a person can do is more important than what degree they obtained.”*

**Nintendo over logic:** *“The key to winning in Nintendo is constant, persistent trial-and-error to discover the hidden doors. The fastest way to winning is through losing, since each loss is a learning experience. [...] It’s no wonder manuals or instruction sets aren’t used by today’s students—they aren’t needed. When handed a digital gadget, these students turn it on, push the buttons, try the knobs.”*

**Multitasking way of life:** *“Many young people today [...] don’t concentrate on one activity at a time. [...] The competition to attract an individual’s attention is one of the most challenging issues of the twenty-first century.”*

**Zero tolerance for delays:** *“It is human nature to want the immediate gratification of our desires, not just physical but informational as well. [...] Voice mail and e-mail are more efficient and less formal, which implies “I want a quick response”.”*

Prensky [Prensky, 2001a] further states:

**Graphics are preferred:** *“Digital natives are used to receiving information really fast. [...] They prefer graphics before text rather than the opposite.”*

McEuen et al. [McEuen, 2001] describe three essential elements of information technology fluency that are co-equal, each reinforcing the others:

1. **Foundational concepts:** *“The basic principles and ideas of computers, networks, and information, which underpin technology. Concepts explain the how and why of information technology, and they give insight into opportunities and limitations.”*
2. **Contemporary skills:** *“The ability to use today’s computer applications, enabling people to apply information technology immediately. Most importantly, skills provide a store of practical experience on which to build new competence.”*
3. **Intellectual capabilities:** *“the ability to apply information technology in complex, sustained situations, encapsulating higher level thinking in the context of IT. Capabilities empower people to manipulate the medium to their advantage and to handle unintended and unexpected problems as they arise.”*

McEuen as well as several studies on use and ownership of emerging technologies by young people [Kvavik, 2005], [Oliver and Goerke, 2007], [Kennedy et al., 2006] showed that students mostly possess contemporary skills. Students reported the use of computers for writing documents and e-mails, followed by surfing the Internet for pleasure, and for classroom activities. They rarely use technology for creating or editing video and audio as well as for creating Web pages. Kvavik [Kvavik, 2005] further states:

*“The interviews indicated that students are skilled with basic office suite applications but tend to know just enough technology functionality to accomplish their work; they have less in-depth application knowledge or problem solving skills.”*

## 2.4 Pedagogical Principles

Savery et al. [Savery and Duffy, 1996] summarize several instructional principles based on constructivist philosophy [Piaget, 1952], [Piaget, 1954] that can guide one in the practice of teaching and design of learning environments:

**Anchor all learning activities to a larger task or problem:** *“That is, learning must have a purpose beyond, “It is assigned”. [...] The purpose of any learning activity should be clear to the learner. [...] the important issue is that the learner clearly perceives and accepts the relevance of the specific learning activities in relation to the larger task complex.”*

**Support the learner in developing ownership for the overall problem or task:** *“Instructional programs typically specify learning objectives [...], assuming that the learner will understand and buy into the relevance and value of the problem. Unfortunately, it is too often the case that the learners simply do not accept the goal of the instructional program [...]. The goals of the learner will largely determine what is learned. Hence it is essential that the goals the learner brings to the environment are consistent with our instructional goals. [...] we may solicit problems from the learners and use those as the stimulus for learning activities. [...] initiate questions (puzzlements) that can serve as the foundation of learning activities [...] the strategy is to define a territory and then to work with the learner in developing meaningful problems or tasks in that domain.”*

**Design an authentic task:** *“[...] the learner should engage in scientific activities which present the same “type” of cognitive challenges. The goal in designing the learning environment is to insure that the cognitive demands, i.e. the thinking required, is consistent with the cognitive demands in the environment for which we are preparing the learner.”*

**Give the learner ownership of the process used to develop a solution:** *“If the learner is engaging in learning activities because the teacher or the instructional program direct them to, then they are no longer relating the specific learning activities to that larger task. [...]”*

**Encourage testing ideas against alternative views and alternative contexts:** *“[...] The importance of a learning community where ideas are discussed and understanding enriched is critical to the design of an effective learning environment. [...]”*

Problem-Based Learning (PBL) is a student-centered methodology in which students acquire knowledge through the experience of problem solving. Savery further suggests:

*“As with any instructional model, there are many strategies for implementing PBL, but they all have in common the use of authentic problems and the stimulus for an organizer of learning activities and the learners work in small collaborative groups.”*

## 2.5 Educational Robotics

Similarly to personal computers which found their way into households and offices since the 1980's, robots have been made affordable for personal use approximately from the mid 1990's.

While robotic technology seemed out of reach due to its expensive price and required expertise and was therefore only targeted for industry, a number of manufacturers started to produce small personal robots commercially available for everyone. Soon innovative educators started to use these personal robots in the classroom. Since then robots became a more and more popular learning tool and the number of cheap, off-the-shelf educational robot kits exploded. There are several reasons why robots are such a popular learning tool in education. Since the 1980's universities notice a constant enrollments decline in technology related disciplines [Patterson, 2005], [Rocard et al., 2007]. The consequent lack of workforce in a society that on the other hand demands more and more on skilled people in these disciplines caused stakeholders to take measures. Institutions to promote science and technology in schools were founded and a variety of attractive out-of-school activities were created. Robotics seems to unify all required skills such as problem solving, logic reasoning, computer science and engineering as well as team work. Further, robots are seen as appealing to students since they are perceived as futuristic and fun. Another important reason that speaks for the use of robots in class is its hands-on approach, which facilitates the application of pedagogical principles such as constructivism, problem-based learning as well as collaborative learning (section 2.4).

The emerging research field of “educational robotics” addresses educational questions as well as technological problems. From the respective literature two main areas can be distinguished: “robotics in education” and “education in robotics” [Alimisis, 2012]. Robotics in education refers to the use of robots as a tool to teach a specific subject, for instance to teach programming in an introductory computer science class. Education in robotics aims at teaching robotics (mechatronics, control theory etc.) and tries to find optimal teaching styles in order to maintain motivated and successful learning students. Often, these two areas are interwoven and a clear distinction cannot be drawn, since working with robots most likely involves learning about mechatronics even though the educational goal lies somewhere else.

### 2.5.1 Educational robotics initiatives

A number of initiatives exist which aim to promote STEM (science, technology, engineering and mathematics) disciplines among young people. We mention here only a number of the most prominent ones in Europe and in the USA.

#### Roberta

This European initiative aims at raising especially girls' interest in STEM disciplines at primary and secondary school level. Roberta is a well developed educational concept that provides gender-balanced didactic material, class concepts, as well as teacher training. To maintain a certain quality, only Roberta certified teachers are allowed to conduct Roberta courses and become part of the Roberta center network in Europe. Roberta courses use LEGO Mindstorms as a robotic platform, however, the course concept is flexible enough to switch to another platform if necessary in the future [Bredenfeld and Leimbach, 2010].

#### FIRST (For Inspiration and Recognition of Science and Technology)

This organization was founded in 1989 in the USA with the aim to inspire young people to pursue a career in science and technology. FIRST created several international robotics competitions<sup>6</sup> for students between 6–18 years. The competitions use mainly the LEGO Mindstorms and Tetrix construction kits (FIRST LEGO League, Junior FIRST LEGO League and FIRST Tech Challenge), but also custom robots are being built by the students (FIRST Robotics Competition).

---

<sup>6</sup><http://www.usfirst.org/roboticsprograms>



### **IPRE (Institute for Personal Robots in Education)**

This institute is a joint effort between Georgia Tech and Bryn Mawr College in the USA sponsored by Microsoft Research. The use of robots as a learning tool for computer science education is tested and evaluated. In its curriculum the Parallax Scribbler robot is used with a custom add-on board. A Python programming infrastructure (Myro) has been developed to support the curriculum's goals [Balch et al., 2008]. The IPRE<sup>7</sup> course concept has been implemented in several introductory programming classes for computer science undergraduates [Markham and King, 2010], [Blank and Kumar, 2010].

### **CEEEO (Center for Engineering Education and Outreach)**

This center of Tufts University, Boston, USA is supporting efforts to integrate engineering into kindergarten and primary school education. The CEEEO<sup>8</sup> consists of four departments: outreach, products, research, and workshops. The outreach and workshop initiatives help teachers to integrate engineering into existing lesson plans as well as provide teacher training. The center's research projects focus on improving technological literacy for all ages at all levels of education, including teachers themselves. They are further involved in product development in collaboration with the LEGO Group and National Instruments (e.g. Robolab (for LEGO), LabView Education Edition, etc.).

### **KIPR (KISS Institute for Practical Robotics)**

KIPR<sup>9</sup>, founded 1994 in the USA, is a provider of educational robotics programs. It organizes the Botball robot competition, robotics camps, conferences and further produces and distributes a variety of robot kits and materials.

### **TERECop (Teacher Education on Robotics-Enhanced Constructivist Pedagogical Methods)**

This European research project (2006-2009) aimed to develop a framework for teacher education courses in order to enable teachers to give robotics-enhanced constructivist classes in schools. Further, the practical implementation of the selected tools both in training courses and in real classrooms are being tested and evaluated. [Alimisis et al., 2010], [Alimisis et al., 2007]. This European project ended, but post-TERECop activities are still ongoing such as teacher training courses and the annual international workshop "(TRTWR) Teaching Robotics Teaching with Robotics – Integrating Robotics in School Curriculum".

### **Robot competitions**

A number of international robot competitions for students emerged within the last decade. The most famous are: RoboCupJunior<sup>10</sup> [Visser and Burkhard, 2007], [Sklar et al., 2003], [Hofmann and Steinbauer, 2010], EUROBOT<sup>11</sup>, FIRST LEGO League<sup>12</sup>, Botball<sup>13</sup>, Robolympics<sup>14</sup> (Switzer-

<sup>7</sup><http://www.roboteducation.org>

<sup>8</sup><http://www.ceeo.tufts.edu>

<sup>9</sup><http://www.kipr.org/>

<sup>10</sup><http://www.robocup.org/robocup-junior/>

<sup>11</sup><http://www.eurobot.org>

<sup>12</sup><http://www.firstlegoleague.org/>

<sup>13</sup><http://www.botball.org/>

<sup>14</sup><http://www.robolympics.ch>

land), World Robot Olympiad<sup>15</sup> (mostly Asia), and Robotour<sup>16</sup>.

Based on this review we see that there are a lot of initiatives on each level of education. All institutions are interested in studies about the impact of the activities on students' future career choices. Some initiatives already carry out their own studies, but so far only with limited coverage. There is a great potential for educational robotics research to conduct more long-term studies in close collaboration with these initiatives.

## 2.5.2 Educational robotics at all levels of education

Educators realized the influence of an early familiarization of science and engineering in the decision making of a student. Robots have therefore conquered classrooms at all educational levels. In what follows we mention some related literature for each academic level.

### Kindergarten

Engineering professions suffer from a negative stereotype, especially among female students. The later in the school career of a student the more established are these stereotypes. For that reason first attempts are being made at the earliest stage of education, kindergarten, where no differences between girls and boys can be observed in approaching robots. For these young children the Bee-Bot<sup>17</sup> is usually used. [Stoeckelmayer et al., 2011], [Pekárová, 2008], [Bers, 2008], [DeMichele et al., 2008].

### Primary and secondary school

Robots in class are often used as a tool to teach traditional disciplines in natural sciences such as physics in a more appealing way [Church et al., 2010]. To avoid the impression that one has to learn for learning's sake, students are engaged in real world science in order to develop conceptual understandings of physics principles through the process of investigation, data analysis, engineering design, and construction. Students exhibiting enthusiasm and ownership when allowing them to solve problems where the answer is not already "known". Therefore, students are more confident and further improve their problem-solving and teamwork skills. There are a number of educational robotics activities in primary and secondary schools in Europe as well as in the USA. Some are initiated by motivated and innovative teachers, others in collaboration with initiatives and research institutions. The most prevalent robotic platform for these academic levels is the LEGO Mindstorms kit. There is a large number of literature covering teaching experiences in schools, such as [Frangou et al., 2010], [Barbero et al., 2011], [Kabátová and Pekárová, 2010], [Pittí et al., 2010], [Nourbakhsh et al., 2005], [Petre and Price, 2004], [Bredenfeld and Leimbach, 2010].

### Undergraduate level (universities)

At undergraduate level, educational robotics is often used as an introductory course for programming. On the one hand to provide easy access to the fundamental logical building blocks of a programming language (e.g. loops and if-else conditions) on the other, to motivate those freshman for computer science which have not made up their mind yet about their major. It is often argued in the literature that robots provide intrinsic motivation to explore the science and engineering behind it. The students continue working for reasons that are different from those traditionally

<sup>15</sup><http://www.wroboto.org>

<sup>16</sup><http://robotika.cz/competitions/robotour/en>

<sup>17</sup><http://www.terrapinlogo.com/bee-botmain.php>

identified. Reasons mentioned are fun, curiosity, and showing off to friends and family [Blank, 2006]. The students initiate the debugging procedure naturally in order to understand and fix a program. This is triggered by their own perception because the robot does not perform as they wanted, rather than by the teacher telling them to do so [Blank, 2006]. Programming robots gives immediate feedback and satisfaction to the students [Markham and King, 2010]. For further reference some related publications in the context of undergraduates education are: [Wyffels et al., 2010], [Hlinovsky and Polcar, 2010], [Apiola et al., 2010], [Imberman and Klibaner, 2005], [Wyffels et al., 2011], [Petrovic et al., 2010].

### 2.5.3 Educational robotics – some critical reviews

Fagin et al. point out the lack of quantitative studies in the literature that assess how robots affect learning [Fagin and Merkle, 2002]. To fill the gap, they were conducting one of the few quantitative studies in an undergraduate introductory computer science class where one class was taught using robots and the control group was taught in a traditional way. They found that the test scores of the students were actually lower in the robotic group, and the use of robots did not have any effect on students' choice of discipline. The conclusion, however, was that the most significant factor that accounts for this was the lack of a simulator for the robotic platform. The students could not take the robot for homework and therefore had to exploit the designated class time to practice programming whereas the other students in the traditional curriculum did not have this constraint.

Kay addresses the question whether robots as a recruitment tool in computer science is simply misleading [Kay, 2010]. Robots are being used to make the introductory programming class more exiting, but when students move on with their computer science studies, they are unlikely to see robots again in the classroom (until they might take advanced courses in AI or robotics).


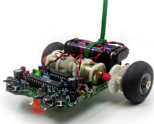

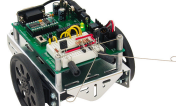




Bredenfeld et al. review the current European educational robotics initiatives and point out general shortcomings [Bredenfeld et al., 2010]. They evaluate the diversity of activities in terms of target audience (academic level, age), the kind of activities (workshops, competitions, conferences), dissemination (local level, national, international) and networks (associations). The research in the field of educational robotics is rather new and currently only a few conferences are fully dedicated to that topic (e.g. RiE (Robotics in Education) or the "Teaching Robotics Teaching With Robotics" workshop). Nevertheless, the fact that the number of conferences and workshops covering educational robotics topics is growing, shows an increasing interest for this research. A rich collection exists on related literature at an international level. Bredenfeld, however, points out the characteristics of these publications: they are usually related to a single institution or small region, they describe examples of small educational projects, results are justified by a very small population from a single class, publications are mostly composed by engineers but rarely in collaboration with researchers from related fields such as pedagogy or psychology. Further, the approaches usually are not condensed to a common standardized methodology, they just describe yet another robotic platform. In general, publications in educational robotics often lack scientific evaluations and examinations of long term impact. Those educational robotics publications that include evaluation studies conclude often solely that the robotics class was motivating and fun. Since to date, there is no long term study about the actual impact of these activities, one can argue that fun is not a serious educational objective and any new teaching tool or activity in the classroom is initially motivating. Robotics is not a compulsory subject in the school curriculum, but rather an optional or one-time event. Therefore, it is hard to conduct long term studies to investigate if this motivation wears off after some time while working with robots.

FIRST works closely together with the Center for Youth and Communities at Brandeis University to provide tangible evidence of the impact of their activities on students' education and career choices. After some preliminary studies in 2005 [Melchior et al., 2005a], [Melchior et al.,

2005b], they launched in 2012 a longitudinal study which examines the impact of the three FIRST robotics competitions on the students over a period of up to five years. They are planning to compare changes in key outcomes to those of a comparison group of young people who are not involved in FIRST. Key outcomes for the study include interest in STEM and STEM-related careers, college-going, pursuit of STEM-related college majors and careers. Results from the preliminary studies indicate that FIRST alumni have gone on to college, pursued educational programs in science and engineering, and maintained a degree of active involvement in the community at higher rates than students from a comparison group with similar demographic and academic backgrounds. Further, alumni have selected engineering as a major in college at a rate that is roughly seven times the national average and more than three times that of students with similar backgrounds in math and science in high school.

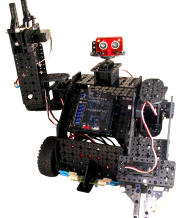


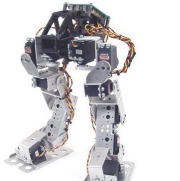
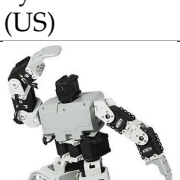
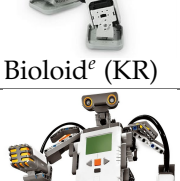
Other institutions involved in educational robotics activities such as the RoboCupJunior started to conduct more long-term studies [Kandlhofer et al., 2012]. The quantitative and qualitative evaluations are planned to span over 5-7 years and track the careers of RoboCupJunior members.

**Table 2.1:** A collection of fixed educational robot platforms

Product	Description	Price
 Bee-bot <sup>a</sup> (US)	Car-like vehicle for young children. Directional keys to enter and save commands to move forwards, back, left, and right. The robot moves in fixed 6 inch steps and turns in 90 degrees angles.	EUR 54
 ASURO <sup>b</sup> (DE)	Car-like vehicle. Assembly by soldering, programmable with AVR C/C++. 2 DC motors, Atmel AVR 8bit microcontroller, light sensor for line following, 6 tactile sensors, wheel encoders, infrared communication. Extras: ultrasonic module, display, temperature sensors, educational material.	EUR 70
 Thymio <sup>c</sup> (CH)	Car-like vehicle. 2 DC motors with encoders, temperature sensors, line following sensors, 5 proximity sensors, 3-axis accelerometer, infrared communication, microphone, memory card slot, speaker, a number of LED's. Graphical (Aseba) and textual programming.	EUR 80
 Boe-bot <sup>d</sup> (US)	Car-like vehicle with on-board breadboard. BASIC Stamp microcontroller, tactile sensors, light sensors, IR proximity, speakers, RS232 communication, textual programming with PBASIC. Extras: additional sensors.	EUR 125
 RoombaCreate <sup>e</sup> (US)	Car-like vehicle. Programmable version of the vacuum cleaner robot (vacuum cleaner functionalities removed). 25 pins for I/O, bumper sensors, wheel drop sensors, proximity sensors, serial communication.	EUR 170
 ePuck <sup>f</sup> (CH)	Small car-like vehicle, 2 stepper motors, 16 positions rotating switch, 8 IR proximity sensors, 3-axis accelerometer, 3 microphones, speaker, 640x480 color pixels camera, 8 LED's, dsPIC microcontroller, RS232 and Bluetooth communication. Extras: Extension possibilities, simulator, remote control.	EUR 700
 Robotino <sup>g</sup> (DE)	Car-like vehicle. Embedded PC running Linux, omnidirectional wheels, bumper sensors, 9 IR proximity sensors, VGA color camera, wheel encoders, power measurement, WLAN and USB communication, graphical programming, or textual using C/C++, Java, LabView, Simulink, etc. Extras: additional sensors such as laser, gyroscope, additional interfaces, educational material.	EUR 5200
 NAO <sup>h</sup> (FR)	Humanoid robot. 25 degrees of freedom, WLAN and LAN communication, 2 cameras, 4 microphones, ultrasonic sensor, 2 IR transceivers, inertial measurement unit, 9 tactile sensors, 8 pressure sensors, voice synthesizer, 2 speakers, Intel ATOM 1,6 GHZ CPU, simulator, Linux with NAOqi middleware, graphical programming, Python, C++, Java, Matlab.	EUR 12500

<sup>a</sup><http://www.terrapinlogo.com/bee-botmain.php><sup>b</sup>[http://www.arexx.com/arexx.php?cmd=goto&cparam=p\\_asuro](http://www.arexx.com/arexx.php?cmd=goto&cparam=p_asuro)<sup>c</sup><https://aseba.wikidot.com/en:thymio><sup>d</sup><http://www.parallax.com/><sup>e</sup><http://www.irobot.com/create><sup>f</sup><http://www.e-puck.org><sup>g</sup><http://www.festo-didactic.com/ch-de/lernsysteme/robotino-forschen-und-lernen-mit-robotern/><sup>h</sup><http://www.aldebaran-robotics.com>

**Table 2.2:** A collection of flexible educational robot platforms

Product	Description	Price
 Multiplo <sup>a</sup> (AR)	Arduino based controller, DC motors, servo motors, remote control, ultrasonic distance sensors, IR sensors, light sensors, USB communication. Mechanical construction: plastic parts (plates, hinges, wheels, gears, pulleys) are being snapped together. Graphical (Minibloq) or textual programming using C/C++.	starting from EUR 110
 Topobo <sup>b</sup> (US)	Construction kit with kinetic memory. The motion is recorded by rotating the motors manually. The stored movements and then played back. Mechanical construction: plastic parts being snapped together.	starting from EUR 120
 RoboRobo <sup>c</sup> (KR)	Atmel AVR microcontroller, DC motors, servo motors, speakers, light sensors, IR sensors, tactile sensors, serial communication. Mechanical construction: wheels, metal plates, shafts, pulleys, etc. screwed together. Graphical (ROGIC) and textual programming using C.	starting from EUR 148
 Lynxmotion <sup>d</sup> (US)	Atmel AVR microcontroller, DC motors, servo motors, IR proximity sensors, accelerometers, pressure sensors, bumpers, remote control, etc. Mechanical construction: wheels, aluminum grippers, servo frames, etc. screwed together. Graphical (e.g. Flowbotics) and textual programming using C.	starting from EUR 230
 Bioloid <sup>e</sup> (KR)	Atmel AVR microcontroller, servo motors and sensors (IR proximity, gyroscopes) are connected in a daisy chain, remote control, serial communication, ZigBee. Mechanical construction: plastic and metal parts, plates, servo frames screwed together. Graphical (RoboPlus) and textual programming using C.	starting from EUR 270
 LEGO Mindstorms NXT <sup>f</sup> (DK)	Controller (NXT) brick supporting 3 motors and 4 sensors using proprietary connectors. Basic set includes: DC motors with encoders, tactile sensors, ultrasonic sensors, light sensors, microphone. A number of additional sensors are available as well as third-party add-ons. Mechanical construction: plastic LEGO bricks (wheels, bars, shafts, etc.). Graphical (RoboLab, LabView for Education, Mindstorms for Education) and textual programming using C, Java, etc.	starting from EUR 350

<sup>a</sup><http://multiplo.org><sup>b</sup><http://topobo.com><sup>c</sup><http://www.roborobo.co.kr><sup>d</sup><http://www.lynxmotion.com><sup>e</sup>[http://www.robotis.com/xe/BIOLOID\\_main\\_en](http://www.robotis.com/xe/BIOLOID_main_en)<sup>f</sup><http://mindstorms.lego.com>

# Design Approach

In this chapter we address the conceptual research questions 1–2 of chapter 1.

1. *How should a platform be designed to enable versatility, adaptability, and scalability?*
2. *How should a platform be designed to explore principles of biologically inspired robotics and embodied artificial intelligence?*

To find answers for these questions, we firstly describe in section 3.1 our experiences with teaching robotics to students. This is followed by a discussion of the drawbacks of current educational robot kits. Based on these findings we define in section 3.2 design ideas for a new toolkit with the aim to meet the requirements of the above listed research questions.

## 3.1 The Need to Develop a Novel Robotic Toolkit

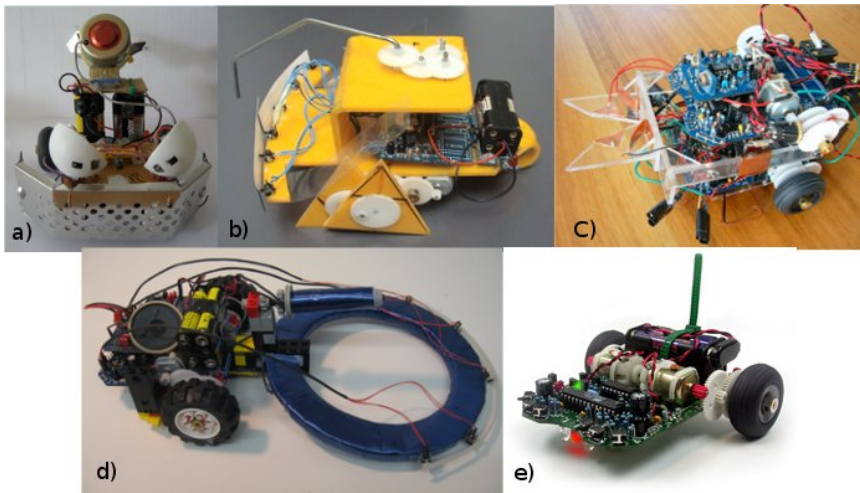
The endeavor to develop an educational robot kit from scratch emerged from our need for an adequate teaching platform. The AILAB has a tradition in teaching biologically inspired robotics and embodied artificial intelligence to students with different backgrounds. We have always emphasized the importance of conducting hands-on workshops to increase motivation and learning productivity. We tried out a number of different educational robot kits in the past. Our teaching activities and experiences are summarized in section 3.1.1, the consequently found drawbacks of current educational robot kits are discussed in section 3.1.2.

### 3.1.1 Experiences with teaching robotics

The computer science undergraduates at the Department of Informatics, University of Zurich have only limited knowledge in electronics and robotic hardware. For that reason, we held hands-on robotics seminars in the last couple of years in order to provide an opportunity for them to “get their hands dirty”. In these seminars the students had to assemble an Asuro<sup>1</sup> robot from scratch (by soldering all electronic components to the PCB) and modify the default body shape as much as possible. Additionally, they had to program the robot to solve their custom chosen assignments. Figure 3.1a, b, c, d show some robots built by the students in the context of the AI seminars. They changed the default body of the robot (Figure 3.1e) by editing and extending its electrical schematics as well as by attaching different structural components.

---

<sup>1</sup>[http://www.arexx.com/arexx.php?cmd=goto&cparam=p\\_asuro](http://www.arexx.com/arexx.php?cmd=goto&cparam=p_asuro)



**Figure 3.1:** A collection of robots built by undergraduates of the University of Zurich in the context of the AI seminar. a) Firebot by Flurin Casanova, Roman Haag, Andres Petralli, b) legged robot with arm by Martin Huber, Yannick Widmer, c) ASURO<sup>2</sup> by Raphael Blatter, Larissa Scherrer, Stefan Hutter, Thierry Bourquin, d) metal detector by Marcelo S. Zanetti, e) the default ASURO robot.

We were further involved in a robotics class in the context of a teacher education program (MAS Ergänzungsfach Informatik an Gymnasien) [Assaf and Pfeifer, 2011b]. The robotics class was part of a two-year master's degree program that aims at educating upper secondary school teachers with different backgrounds in computer science, a discipline that is not yet a mandatory part in the school curriculum in Switzerland. The teachers were engaged in two robotic competitions with the goal to introduce them to robotic hardware and software such that they will be able to design their own class syllabus. For one competition we used the LEGO Mindstorms platform. The task emphasized the importance of morphology (body shape and material properties) of a robot. The teachers had to achieve stable and fast locomotion by only changing the body of the robot by leaving the control fixed (constant forward rotation of two motors). The other competition involved programming a custom Arduino based robot.

We additionally carried out a number of workshops with students from primary to secondary school levels at special open days or events. Based on all our teaching experiences we found that both teachers and students were overall very engaged in hands-on robotic exercises. They approached the assignments (predefined or self defined) initially very motivated and tend to aim high. They were, however, not aware of the fact that working with robots can be very problematic due to noise of sensors, inaccurate motors etc. These experiences are on one hand important and valuable on the other hand can easily lead to frustration. The teachers and students could rarely solve their self defined tasks because of this lack of experience. We learned to point out that fun and frustration go hand in hand while working with robots. Therefore, assignments should not be designed too ambitiously and complexity should rather be increased step by step in order to ensure motivation through the feeling of accomplishment. Another observation was that unless the user was an experienced programmer, most of the time had been spent on debugging source code (e.g. finding missing semicolons, understanding why variables have to be declared with the correct data types, etc.). This was also a cause for frustration and decreasing motivation. We further found the tendency of students in trying to solve a problem through programming. It is, however, sometimes more efficient and easier to approach a problem by changing the morphology of a robot. We aimed to show that with the LEGO Mindstorms morphology exercise described above. Braitenberg vehicles [Braitenberg, 1986] was also usually a very popular exer-



cise. Especially the teachers liked the simplicity of these exercises and the message behind them. We therefore created and conducted a large number of exercises that demonstrate the principles of embodied AI.

More details about our teaching experiences in the teacher education program can be found in the following publication (Appendix A) [Assaf and Pfeifer, 2011b]:

Dorit Assaf and Rolf Pfeifer

*Robotics as Part of an Informatics Degree Program for Teachers*

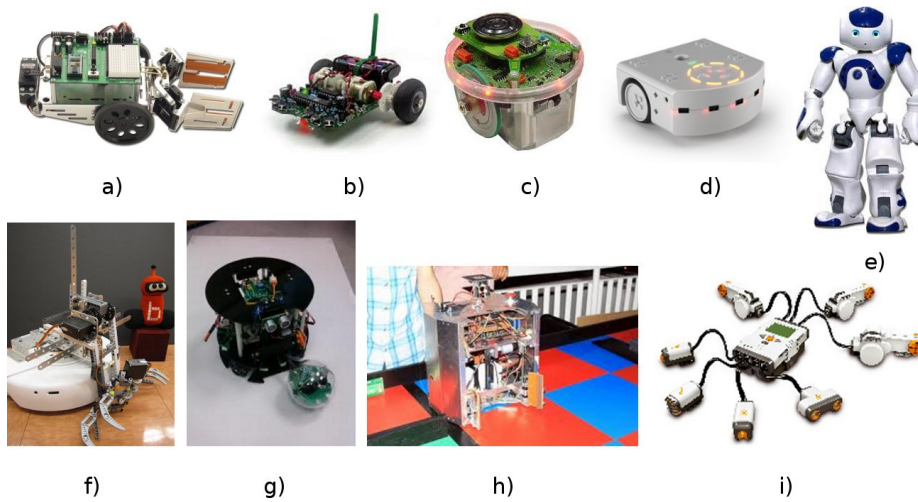
**Abstract** *This paper describes two robot competitions that took place within a robotics class for teachers. The robotics class was part of a two-year master's degree program that aims at educating upper secondary school teachers of different backgrounds in informatics, a discipline that is not yet a mandatory part of the school curriculum in Switzerland. The aim of these robot competitions was to familiarize the teachers with robotic hardware and software such that they would be able to design their own informatics class syllabus. We describe the robotic platforms used, the competitions, their aims and results. Furthermore, we address the question whether robots are a suitable tool for teacher education in informatics.*

### 3.1.2 The drawbacks of current educational robot kits

A more detailed review of educational robot kits can be found in section 2.2.1. We summarize here again shortly the most prevalently used toolkits in education in order to clarify our point of view.

Figure 3.2 shows a collection of typical educational robot platforms that are used worldwide in educational robotic activities. Robot platforms such as the boe-bot (Figure 3.2a), Asuro (Figure 3.2b), e-puck (Figure 3.2c), Thymio II (Figure 3.2d) and Nao (Figure 3.2e) are limited in extension and modification possibilities due to their default shape. From the more flexible platforms in respect to body shape, LEGO Mindstorms (Figure 3.2i) is the most prevalently used one. However, a frequently addressed issue with LEGO Mindstorms is its expensive price and limited scalability (it only allows up to four sensors and three actuators to be connected to the controller). The pre-defined body shapes and limited scalability might explain the observation, that the robots used in competitions are usually to a large extent custom built (Botball Figure 3.2f, RoboCup Figure 3.2g, EUROBOT Figure 3.2h). Some competitions use one platform exclusively like LEGO Mindstorms in FIRST LEGO League, or Nao in RoboCup Standard Platform League. Overall, the robot characteristics do not differ much in either the custom built robots and those with default body shape: they are usually car-like platforms with grippers and common sensors to detect distance and light. As a consequence, the robot assignments in both competitions and robot workshops are often very similar: line following, obstacle avoidance, navigation in space, collection and transport of objects. As criticized by Bredenfeld et al. [Bredenfeld et al., 2010], robotic platforms or class activities are often designed by engineers with only limited collaboration with the other expert groups, pedagogues and teachers (Figure 1.1). This might explain the above mentioned strong engineering focus in educational platforms or assignments.

Based on our teaching experiences and the above mentioned properties of commercially available robot platforms, we decided to develop a novel toolkit. We do not want the user to be constrained to work solely with classical robots (e.g. car-like vehicles). Students can of course build such a robot but additionally they should be able to explore the concepts of biologically inspired robotics and embodied AI, which involve the synthetic methodology as well as robots incorporating compliant material properties and passive dynamics. Preferably, a user should be completely free to construct any technological device that comes into his mind (of course within the technical constraints of the toolkit at hand). We believe, that these classical robotic platforms together with the typical tasks (line following, obstacle avoidance) may not appeal to a broad audience. Educa-



**Figure 3.2:** A collection of robots used in education and competitions. a) boe-bot, b) Asuro, c) e-puck, d) ThymioII, e) Nao, f) Botball, g) a robot used in RoboCup, h) a robot used in Eurobot, i) LEGO Mindstorms NXT

tional robotics activities to date focus very much on classical top-down engineering approaches: a robot is planned and built out of predefined parts, its control program is implemented, tested, and revised in iterative processes. Students tend to focus on a robot's control to solve a given problem, whereas sometimes it is much more efficient to change a robot's body properties. Current educational robotic platforms and activities are with no doubt valuable for students but may address rather those that are already intrinsically motivated. We believe that it might be appealing to a wider crowd of students to learn about biologically inspired robotics and embodied AI, since these topics involve interdisciplinary research fields such as biology, neuroscience, as well as psychology. Furthermore, we face a general decreasing interest of young people in studying engineering disciplines. Most of the educational robot kits focus on programming. Engineering, however, involves many more skills than solely programming. A toolkit should therefore foster problem solving skills that go beyond programming. Students should be able to explore interdisciplinary topics which include but are not limited to: electronics, mechanical engineering, material science, biology, psychology, etc.

## 3.2 Design Approach for a New Toolkit

The new robotic toolkit should support a variety of teaching activities, especially in the field of biologically inspired robotics and embodied AI. We further emphasize the importance of the interdisciplinary fields involved in an educational robotics project (pedagogy, engineering, teaching practice, Figure 1.1). The design of the toolkit should therefore be highly influenced by following factors: current pedagogical principles (section 2.4), studies about the target audience's learning preferences (section 2.3), the design principles of embodied AI (section 2.1), and our own experiences with teaching robotics to students with various backgrounds (section 3.1.1).

We learned in section 2.3 facts about today's students' technology literacy. Since they are mostly fluent in contemporary skills (using software), the toolkit should therefore give insights about computer hardware (foundational skills) and enable editing and manipulating the system according to their needs (intellectual capabilities)(e.g. building a robot). Based on the students' learning preferences we decided to implement a rapid prototyping platform that allows the user

to realize and test an idea easily and fast. A graphical user interface should be provided as well to ensure user-friendliness and immediate feedback. We follow the open-source philosophy which means that our platform will be released under an open-source license and additionally solely will incorporate open-source hardware and software. Furthermore, to meet the versatility, adaptability, and scalability requirements a cheap, modular design approach needs to be chosen.

To learn about principles of embodied AI and biologically inspired robotics, the focus often lies on material properties of a robot's body. Unconventional properties should be incorporated, such as passive dynamics and soft materials. To achieve that, industrial standards have to be kept where possible. It eases the attachment of arbitrary material that can be found cheaply in any do-it-yourself store. Many other manufacturers intentionally develop proprietary standards for monopoly reasons. We don't support these strategies and therefore aim to ensure compatibility with other robotic platforms. The user should be enabled to customize, merge, and adapt different platforms according to his needs.

The principles of embodied AI demonstrate often emergent behaviors, i.e. the behavior of an individual agent (e.g. robot) does not reflect the overall behavior that is observed in interaction with other agents as well as with the environment. By looking solely at the behavior of an individual ant for instance, it is not possible to predict the emerging behavior (e.g. finding food, building a nest) of a thousand ants interacting with each other and the environment. Similarly, simple motor oscillations can result in complex behavior, due to the physics acting on the body in interaction with the environment. The user should therefore be able to explore behavioral principles easily. The focus should not be on programming the control, rather on tuning parameters of a predefined motor control routine (e.g. oscillation).

### 3.3 Summary

To enable versatility, adaptability, and scalability (research question 1) a modular design approach has to be chosen. The user should be able to build a system using sensors and motors by easily adding the required modules, most preferably with plug-and-play properties. To avoid an explosion of costs by increasing scale and complexity of the system, the modules should be designed cheap, re-usable and generic (e.g. the same module can be used for different purposes).

To enable education in biologically inspired robotics and embodied AI (research question 2), the hardware design should comply with industrial standards to enable the attachment of different unconventional materials. The user should further be able to explore the effect of preprogrammed control routines (e.g. oscillations) on a robot's behavior by simply tuning their parameters.



# The EmbedIT Toolkit

This chapter covers the implementation details of EmbedIT, with the aim to find answers for the technical research questions 3–4 of chapter 1.

3. *How should the platform architecture and its interfaces be designed to meet the requirements of question 1–2?*
4. *Which are the technical challenges in developing a novel platform and what are the bottlenecks of the chosen design?*

The toolkit developed in the context of this thesis consists of hardware (custom-designed embedded components), firmware (control program running on the microcontroller), and a client application. Technical details are provided in [Assaf and Pfeifer, 2011a]. Although the toolkit's main concept remains unchanged since the latter publication, many parts have been improved and new features implemented. This chapter describes the most recent state of the toolkit. For readers interested in the development process of this toolkit, see (Appendix B). We refer to the summary of this chapter for those readers solely interested in the answers of the above listed research questions.

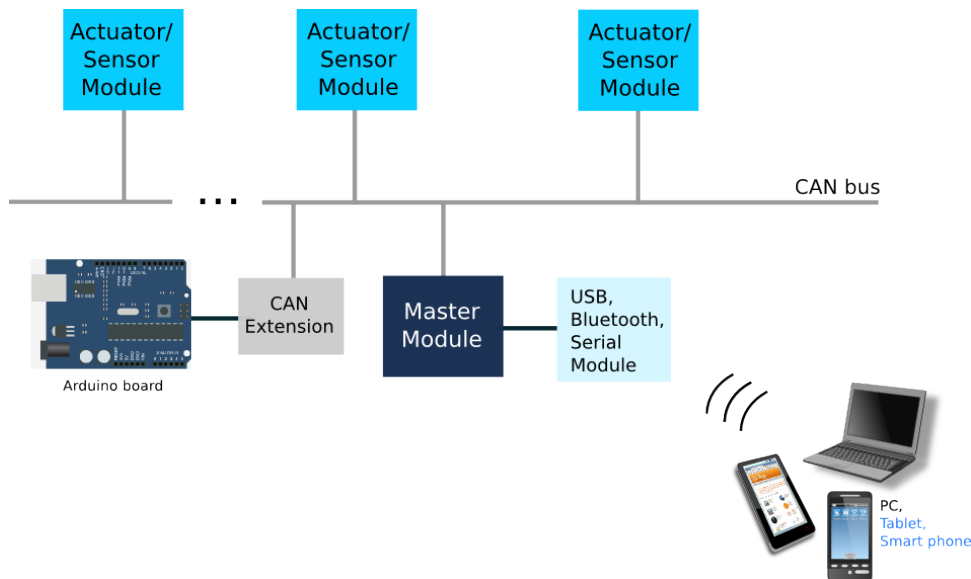
Dorit Assaf and Rolf Pfeifer

### *EmbedIT – an Open Robotic Kit for Education*

**Abstract** *Robots have often been used as an educational tool in class to introduce kids to science and technology, disciplines that are affected by decreasing enrollments in universities. Consequently, many robotic kits are available off-the-shelf. Even though many of these platforms are easy to use, they focus on a classical top-down engineering approach. Additionally, they often require advanced programming skills. In this paper we introduce an open robotic kit for education (EmbedIT) which currently is under development. Unlike common robot kits, EmbedIT enables students to access the technical world in a non-engineering focused way. Through a graphical user interface students can easily build and control robots. We believe that once fascination and a basic understanding of technology has been established, the barrier to learn more advanced topics such as programming and electronics is lowered. Further we describe the hardware and software of EmbedIT, the current state of implementation, and possible applications.*

## 4.1 Hardware and Software

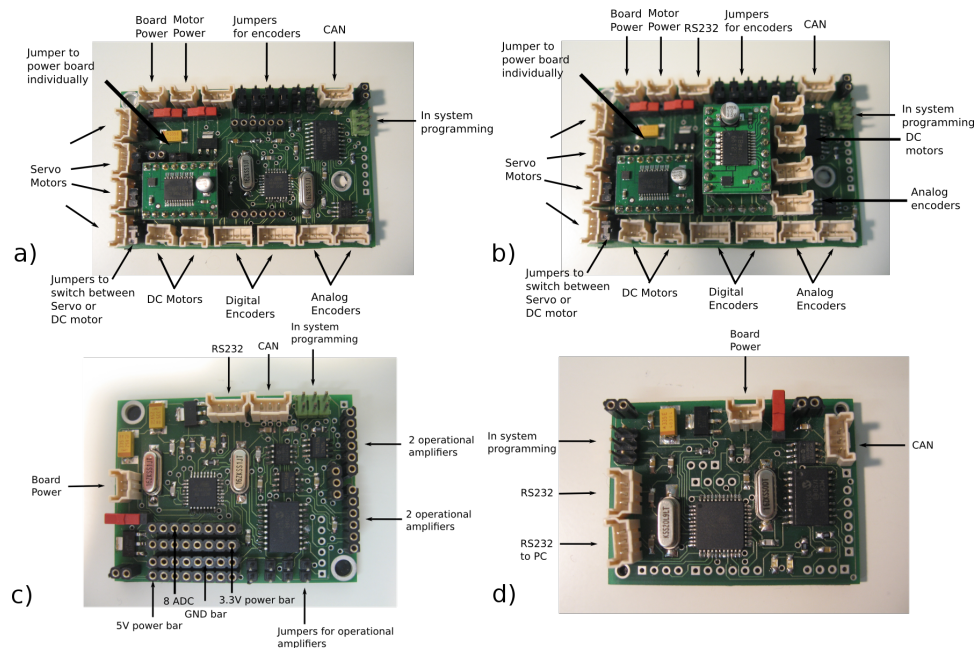
The main hardware components of the EmbedIT platform are three different modules: a master module, a sensor module and an actuator module (Figure 4.1). Each module is an independent



**Figure 4.1:** The EmbedIT hardware system. A master module is attached to the CAN bus and a communication module (Bluetooth, USB or serial) is attached to the master module. The EmbedITApp running on the PC connects to the master module through the communication module. Additionally, actuator and sensor modules are connected to the bus. The actuator module can control either up to four servo motors or up to four DC motors including encoders. One sensor module controls only one type of sensor (e.g. accelerometer, ultrasonic distance sensor, contact sensor, etc.) but up to eight devices (for analog sensors).

entity carrying its unique identification number and control program that serves one specific purpose (control, actuation, sensing). Each module is a small embedded system, i.e. individual printed circuit boards (PCB) equipped with an Atmel AVR 8bit microcontroller. A CAN data bus (two conductor cable) is the connecting piece between the modules. Each module can easily be attached and detached from the bus. The master module, which is a control module, controls the communication traffic on the bus. In addition, the master module serves as an interface to the user (Figure 4.2). A Java application, the EmbedITApp, running on the user's PC connects to the master module through a communication module, that can either be a Bluetooth, USB or serial (RS232) interface. In Figure 4.1 a master module is attached to the CAN bus and a communication module is attached to the master module (communication modules are not attached to the bus directly, they have to be attached to the master module). The EmbedITApp connects to the master module through the communication module. Additionally a number of actuation and sensor modules can be connected to the bus. An actuation module can control up to four servo motors or up to four DC motors including encoders. To switch between servo or DC motor mode, the respective motor types have to be connected to their designated plugs (Figure 4.2a,b). Additionally, the respective microcontroller code has to be uploaded to the actuator module (this can be done through EmbedITApp). Each sensor module supports just one sensor type (e.g. ultrasonic distance sensor, accelerometer, gyroscope, light sensor, etc.) but controls up to eight devices (for analog sensors). To change between sensor types the respective firmware code has to be uploaded to its module. The sensor modules are equipped with operational amplifiers (up to four analog sensors) as well as regulated power supply of 5V and 3.3V (these two voltages are predominantly required for sensors)(Figure 4.2c). A large number of sensors and actuators are supported by the toolkit to present (Figure 4.3).

Figure 4.4 shows the real modules connected together: a Bluetooth module, a master module,



**Figure 4.2:** A detailed description of each EmbedIT module. a) an actuator module supporting either two DC motors or four servo motors, b) an actuator module with DC motor extension attached. This allows to attach two additional DC motors, c) a sensor module, d) a master module.

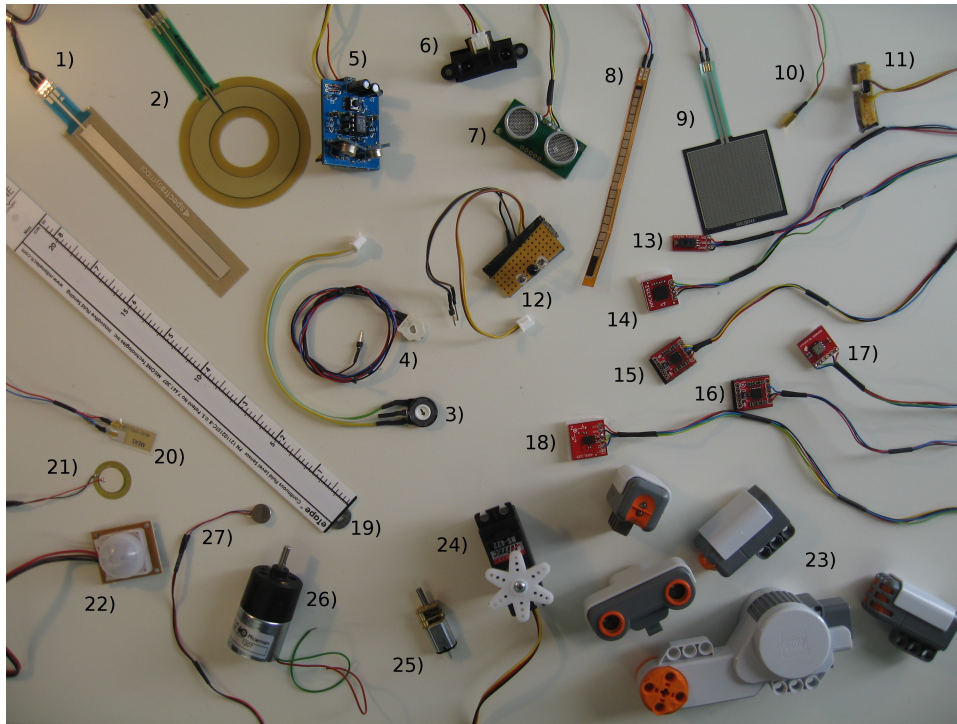
two sensor modules (accelerometer, ultrasonic), the CAN bus, a power supply and an actuator module with one servo attached. Since we use the same microcontroller family than the Arduino boards, they can be easily attached to the bus by using a CAN extension. The LEGO sensors can be integrated in the system as well. Once a connection between the EmbedITApp and the master module has been established, the master module returns the ID and module type (sensor type, actuator type) of all hardware modules that are physically connected to the bus.

Figure 4.5 shows an example of the EmbedITApp where a servo module and four sensor modules (infrared, light, accelerometer, gyroscope) are listed. Each physically available module is represented as a module button showing its ID, module type and icon (upper left column of the application). Clicking on a module's button opens its control panel (upper right column). Here the module can be accessed and controlled directly. The control panel of a light sensor module is displayed in Figure 4.5, upper right column. Up to eight sensors can be controlled with this interface. Sensors can be connected or disconnected. Sensor values can be read according to chosen polling frequencies. The sensor values are displayed either textually or graphically. Additionally, they can be exported as a text document (for later processing).

Figure 4.6 shows the control panel of an actuator module controlling DC motors. The actuator module is able to simultaneously control up to four DC motors. From this control panel commands such as connect/disconnect motors, stop, forwards, backwards can be applied. The speed can be set through a slider or a text box. By pushing the "apply" button, the new control settings are sent and applied directly to the hardware.

Figure 4.7 shows another actuator control panel but this time for servo motors. Here too, commands such as connect/disconnect servos or move to discrete servo positions can be sent. Another possibility is to let the servo motors change positions in a continuous, sinusoidal way (alternating from  $0^\circ$  to  $180^\circ$ ). Different sine parameters such as amplitude (servo angle range),





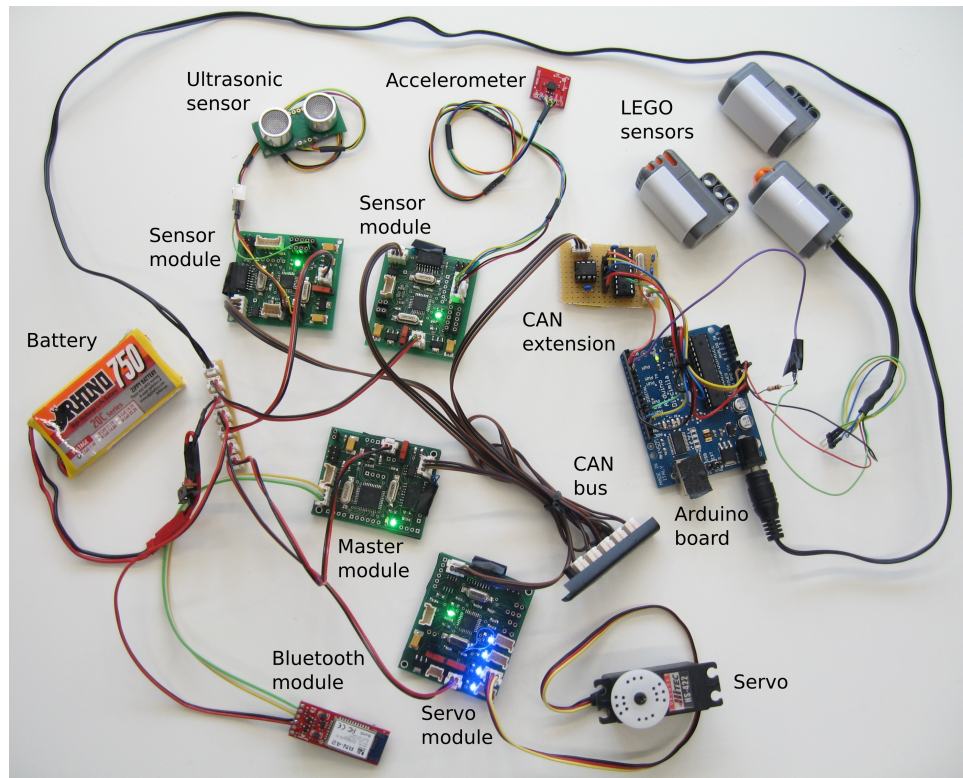
**Figure 4.3:** Sensors and actuators that are supported to date. 1-4: Potentiometers, 5: Infrared temperature sensor, 6: Infrared distance sensor, 7: ultrasonic distance sensor, 8-9: force sensitive resistors, 10-11: switches, 12: light sensor, 13: humidity sensor, 14: compass, 15-16: gyroscopes, 17: barometric pressure, 18: accelerometer, 19: liquid level sensor, 20-21: vibration sensor, 22: motion sensor, 23: LEGO NXT sensors and actuators, 24: standard servo motor, 25-26: DC motors, 27: vibration motor.

offset (shift of the servo middle axis), phase lag (synchronization of all servos) and frequency can be applied. The EmbedITApp sends the commands to the master module which forwards them to the target module on the bus. Servo positions can be changed while dragging the servo position sliders. After pushing the apply buttons on the control panel, the servo positions are changed physically.

By means of the module control panels (Figure 4.5 upper right column, Figure 4.6, Figure 4.7) the user is able to immediately access and control the sensors and actuators. Sensor values are displayed graphically or can be captured and exported for later processing. Motors can be moved to positions in real-time. This enables the user to rapidly test the sensors and actuators in the system, get an idea about how they function and how accurately they measure and actuate. The user can experiment with different motor and sensor types, exchange sensors with other sensor types in order to find out which ones are the most adequate to be used for a specific purpose. Further, the system can be extended easily with more sensors or sensor and actuator types. Simply by adding more modules to the bus the hardware is detected and ready for use.

The graphical interface also lets the user define and save sensor-actuator relationships. Relationships are for instance, if a certain threshold has been passed within a sensor module, the motor reacts accordingly. Conditions are saved on the sensor module (e.g. sensor value > threshold) and high level actions are saved on the actuator module (e.g. turn both motors forwards at a certain speed). Actions are triggered whenever a condition is valid (e.g. sensor value passed threshold). The sensor module then sends a message to all actuator modules dependent of this condition. The defined sensor-actuator relationships are saved on the modules' non-volatile memory (EEPROM),





**Figure 4.4:** Some EmbedIT modules: A Bluetooth module, a master module, two sensor modules (accelerometer, ultrasonic), the CAN bus, a power supply and a servo module with one servo attached. By extending CAN components (CAN extension) to an Arduino board, it can be easily integrated in the system.

which makes them available after a cold hardware restart. The lower panel in Figure 4.5 shows the graphical programming area where two sensor modules (light and accelerometer) and one actuator module (servo motor) are connected together. The user simply drags the modules from the module list (upper left column) and drops them into the programming area. Relationships can be defined through the connection of modules (black lines between modules). At the current state of development, simple relationships can be defined and saved. A continuous mapping of sensor values to actuators is possible as well.

## 4.2 Customizable by the Open-Source Community

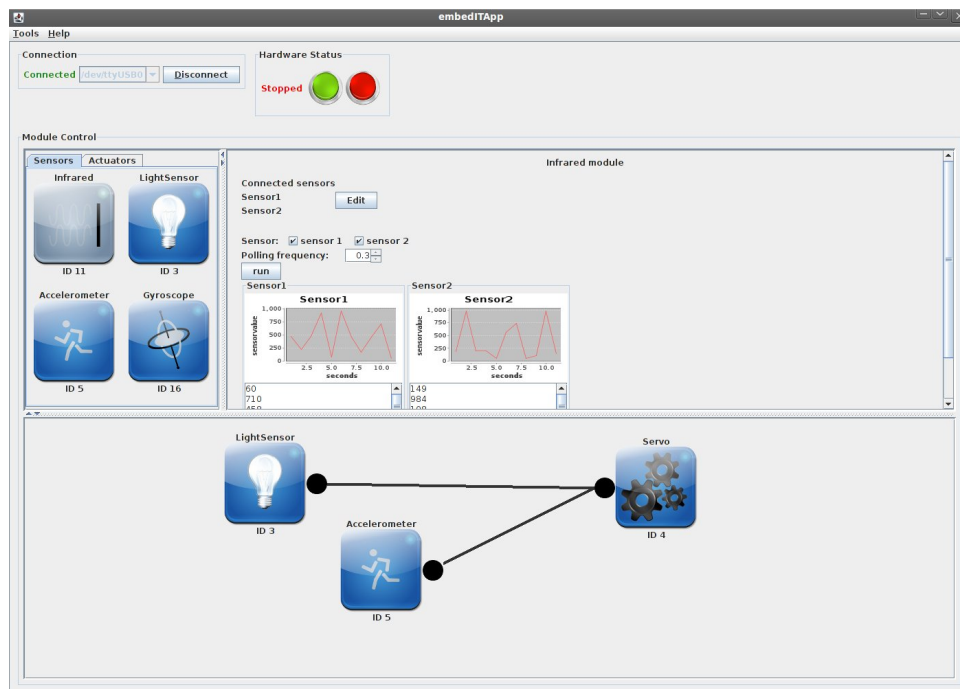
The project incorporates only open-source hardware schematics and software libraries (e.g. RXTX serial library<sup>1</sup>, CAN library<sup>2</sup>, Arduino library<sup>3</sup>). The project plans to release hardware and software as well under an open-source license. The EmbedIT toolkit is accessible and editable at any layer of its architecture (Figure 4.8)

EmbedIT modules are PCB's that are specifically designed for their purpose. Any custom board equipped with a CAN communication interface (e.g. EmbedIT CAN extension of Figure

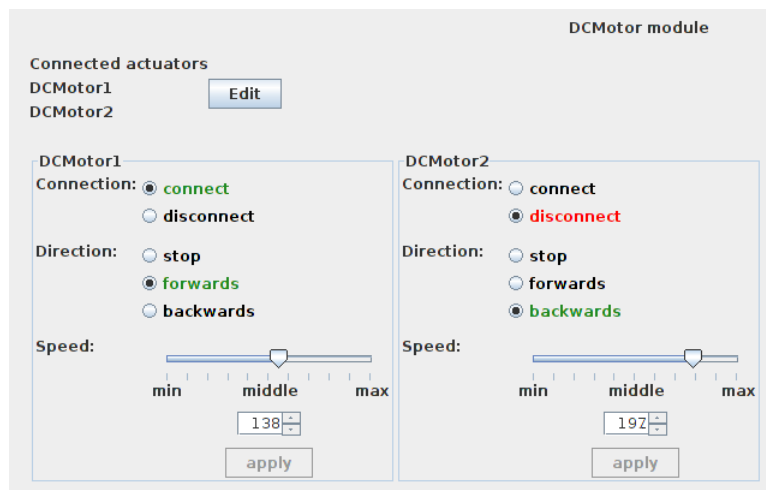
<sup>1</sup><http://rxtx.qbang.org>

<sup>2</sup><http://www.kreatives-chaos.com/artikel/can>

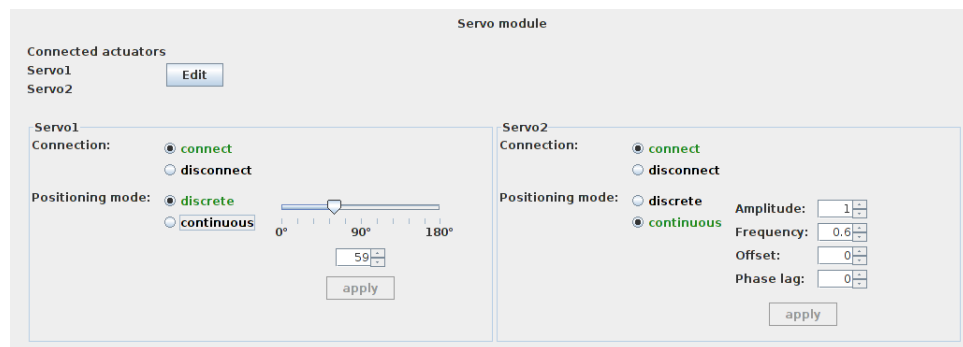
<sup>3</sup><http://www.arduino.cc/>



**Figure 4.5:** An example of the EmbedITApp where a servo module and four sensor modules (infrared, light, accelerometer, gyroscope) are listed. Each available module in the application is represented with its ID, icon and a button (upper left column of the application). To access and control a module directly, the user has to click on a module button to display its control panel (upper right column). In the graphical programming area (lower panel) sensor motor relationships can be defined.



**Figure 4.6:** The control panel of an actuator module controlling DC motors. The actuator module is able to control up to four DC motors at the same time. From this control panel commands such as connect/disconnect motors, stop, forwards, backwards can be applied. The speed can be set through a slider or a text box.



**Figure 4.7:** The control panel of an actuator module controlling servo motors. The actuator module is able to control up to four servos at the same time. From this control panel commands such as connect/disconnect servos, move to specific servo position as well as continuous position change in a sinusoidal way can be applied.

4.4) can be added to the system.

The EmbedITApp application:

- Is programmed in Java and can therefore easily be customized and extended,
- Runs at the moment on a PC, but could be exported to a tablet or smart phone,
- Sends commands consisting of character sequences to the serial communication port (RS232) of its operating system (which are received by the master module). Any custom application could send these commands instead of the EmbedITApp.

The firmware running on a module's microcontrollers:

- Can be altered or extended with own libraries (e.g. Arduino library).
- Can easily integrate additional preprogrammed algorithms or control routines.

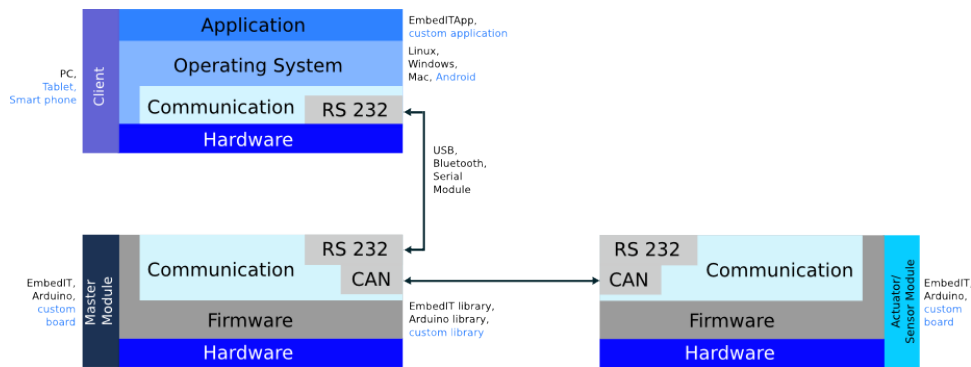
If a user for instance would like to control the motors by means of a joystick, he only has to implement a joystick functionality within EmbedITApp. Another possibility is to design a custom hardware module to be added to the system. The user has to designate a module type and ID to its firmware and implement its support within the EmbedITApp.

We are releasing the software and hardware schematics on the project website<sup>4</sup> together with tutorials and example applications. We hope that a community will emerge in the future which contributes with extensions of the toolkit's functionality.

## 4.3 Mechanical Construction Components

By "mechanical construction components" we refer to structural material consisting of re-usable elements such as strips, plates, angle girders, axles, wheels and gears with nuts and bolts to connect pieces in order to build the body of a robot. We do not custom design any of these components in the scope of this project. We believe that this is not necessary due to the large number of mechanical construction kits available off-the-shelf. Nevertheless, we give instructions on how to attach standard motors and sensors together with the EmbedIT hardware to these mechanical construction kits. Further, we show solutions how to incorporate soft materials and passive dynamics

<sup>4</sup><http://www.embed-it.ch>



**Figure 4.8:** The Architecture of the EmbedIT toolkit. The client (a PC) runs the EmbedITApp, which sends commands consisting of character sequences to the master module through a serial communication interface (RS232). The master module is connected to the other modules through a CAN bus.

into a robot, since no mechanical construction kit to date supports these kind of characteristics (see chapter 3 and 4 of the following publication (Appendix C) [Assaf et al., 2012a]).

Dorit Assaf, Jørgen Christian Larsen, and Markus Reichardt

#### *Extending Mechanical Construction Kits to Incorporate Passive and Compliant Elements for Educational Robotics*

**Abstract** Robots are a popular educational tool to introduce science, technology, and engineering to students. The field of educational robotics is growing and consequently a number of educational robot kits have been developed within the last decade. Our laboratories have a tradition of teaching embodied artificial intelligence and biomechanics to students with different backgrounds. The robots we use both for research and education are usually built incorporating compliant materials as well as passive dynamics. These kind of properties are often not available in classical robot kits or mechanical construction kits. In this paper we describe some of the robots we use for education. So far we built the robots using 3D printing technology which is convenient but too expensive for class use. Our aim is to find cheaper, commercially available solutions. After a short review on educational robot kits and mechanical construction kits we describe interface solutions between several kits. Further we show some solutions to incorporate compliant materials and passive dynamics to traditional mechanical construction kits by using cheap and widely available materials.

## 4.4 Summary

We chose the architecture and interface design of EmbedIT as follows (research question 3): Research question 1 demanded a modular design, where modules can be added easily with plug-and-play functionality. Modules should be cheap, re-usable and generic. We achieved modularity by designing three different modules: the master, actuator, and sensor. The actuator module is designed generic, it can either control servo motors or DC motors. The sensor module provides useful features such as power regulations of 3.3V and 5V. Four analog channels can further be amplified if needed. The module is generic in that sense that solely the respective firmware has to be uploaded to switch sensor types (of course the correct sensor has to be attached to the module). All modules are re-usable and composed of standard electrical components which can be purchased at any supplier in bulk. The plug-and-play functionality was achieved by the CAN bus and the master module managing the bus.

Research question 2 required unconventional material properties and preprogrammed control routines. The unconventional material properties were achieved by finding the simple solutions with spring steel and steel ropes, as described in section 4.3. The preprogrammed control routines were implemented with the decentralized control architecture of the system. A sinusoidal oscillation can be run for instance on a servo motor module. Through the EmbedITApp control window the oscillation parameters can be tuned in real time.

The development of a novel platform is a big challenge and our chosen design has both advantages and disadvantages (research question 4). Firstly, this modular design has never been implemented before in an educational robotic platform. We could therefore not profit from experiences of previous projects. Every design choice has to be weighed up. For example, the more modular a system is designed, the more expensive it gets (the costs of each module adds up), the more difficult it becomes to manage all modules at the same time, the more bandwidth is used on the communication bus. An adequate level of modularity has to be found (e.g. is a group of the same sensors a module or should each sensor be a module?). The choice to provide wireless communication (e.g. Bluetooth) is preferred, but it becomes more difficult to handle. The user has to establish a connection first on his specific operating system before it can be used. Similar trade-offs have to be considered with the graphical user interface. The more simplified the interface, the easier it is to use. A too simplified interface, however, reduces the flexibility of the application. The way the sensor-motor relationships are saved in on the one hand very flexible, on the other hand limited. The non-volatile memory (EEPROM) has limited capacity, too many relationships cannot be saved at the current state of development.



# Teacher Interviews

This chapter addresses the following research questions 5–6 of chapter 1 about the teachers:

5. *What are the classroom practices of teachers giving computer science and engineering classes in secondary schools?*
6. *What are the teachers' requirements for educational robotic toolkits?*

The teachers are the practical education experts and the decision makers who will eventually chose a platform and further design and carry out exercises. It is therefore crucial that engineers developing educational platforms collaborate closely together with this expert group (see arrow 5 and 6 in Figure 1.1). For that reason we conducted teacher interviews to get insights about their teaching practices and personal requirements for a toolkit. We asked for example about the key factors that influence the preference of one platform from another, as well as their thoughts and experiences about the current platform they are using. Further, we demonstrated our current development of EmbedIT and asked for their personal opinion.

*Information and Communication Technology (ICT)* is currently the only IT related, mandatory class taught in Swiss secondary schools. The goal is to deliver basic knowledge about office applications such as Microsoft Word, Powerpoint or Excel as well as to impart media literacy. Topics in computer science (programming, algorithms, information theory, computer hardware etc.) are not covered in this curriculum. Switzerland has a lack of work force in IT related professions and faces decreasing interest of students pursuing a career in these disciplines. As a measure, school authorities introduced a new, optional discipline called *Ergänzungsfach Informatik (EFI)* (complementary informatics) a couple of years ago. Students that are interested in computer science can now enroll for this graded class. The curriculum is not fixed which leaves the teachers to choose the actual topics being covered. However, a recommended syllabus exists, composed by the school authority. The problem at present is to find enough teachers that have the expertise to teach this new class. The EFI study program [Assaf and Pfeifer, 2011b] (mentioned in section 3.1.1) targeted teachers that are interested in receiving additional training in computer science and thus are able to teach the EFI class besides their usual disciplines.

To find teachers that are currently giving the EFI class in schools we got back to our contacts from this EFI study program. Since then we were anyhow in touch with a number of teachers, which kept us posted about their endeavors, which borrowed the robots we used during the study program, or just asked technical questions. We sent a query who might be interested in providing class time with their students in order to test our toolkit as well as who would be available for a two hour interview. We appreciated much the positive response. From 33 teachers queried, 9 (27.2%) got back to us and volunteered either to provide class time or to be available for an interview, or both.

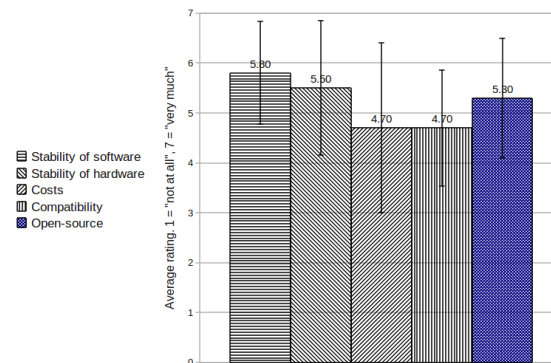
We aimed to learn about the teachers' particular EFI curriculum, their experiences, demands and plans for future classes by means of personal, two hour interviews. Further, we demonstrated our toolkit and asked for their opinion about it. Nine of the ten teachers interviewed participated in the EFI study program and taught the EFI class in schools already for one to five iterations. The teachers were between 32 and 56 years old (average age of 44, two female) and from different places all over Switzerland. Their backgrounds were Physics, Sports, Geography, Mathematics, Media Pedagogy, Biology, Chemistry, Informatics, and Economics.

The EFI class is an optional class which has to compete with other complementary classes that can be chosen by the students (e.g. Sports, Psychology, etc.). We asked about how the EFI class is being "advertised" to encourage students to enroll for it. Normally, all optional classes are being announced together on the dedicated school website. Some students choose a class based on personal recommendations of mature students that took the class in previous years. Other factors are the familiarity of the teacher. From a well known teacher the students have a better idea of what to expect from his class. One teacher mentioned that by being the system administrator of the school, he wasn't known among the students as a teacher and therefore it was harder for him to motivate enough students to join the class. On other occasions the projects of the previous class were presented in an exhibition which helped with the advertisement. Another teacher mentioned that the term "programming" has a negative connotation to some students. They immediately associate Mathematics and complicated things with it. To avoid this daunting effect, the teacher never used the term programming until the students had some first experiences. After a failed attempt to reach the minimum number of students for another teacher's EFI class, he organized an optional, non-graded LEGO Mindstorms workshop with ten students. During that workshop he advertised the EFI class which consequently could take place with twelve students enrolled, half of which attended the previous robot workshop. He even got requests from other students to organize another robot workshop. One teacher tried to motivate the students during her Biology class by mentioning that they are going to play with artificial evolution in the EFI class. Generally, it can be said, that most of the students have neither a specific idea what computer science as a discipline is all about nor do they have previous programming knowledge (except for a few "computer freaks" which can be found in every school). The number of students enrolling for the EFI class varies from school to school every year, but stays averagely at 11 students. The number of female students is low at approximately 17%.

Each teacher creates his individual syllabus with only limited collaboration with other teachers. And a great variety of topics in computer science were brought up in the interviews. Among the most frequently mentioned were: Java programming (with Eclipse, Greenfoot, Processing, or BlueJ), algorithms, databases, networks, game programming, robotics with LEGO Mindstorms, artificial intelligence (chess computers, Braitenberg vehicles) and smart phone programming. Additional topics were programming with Python, scratch, Java Script, Html, PHP and C, learning about computer hardware by opening a computer, numerics, Internet and law, encryption and IT security, computability, logic, computer architecture, client/server architecture, simulations and artificial evolution. LEGO Mindstorms was the predominantly mentioned robotics hardware. Two teachers, however, were hooked by Arduino ever since we introduced this platform to them at the EFI study program. They switched from LEGO to Arduino as their preferred teaching platform. This was rather exceptional, since most of the teachers pointed out to be unwilling to deal with low-level electronics and soldering in a class environment.

Since all teachers were familiar with the LEGO Mindstorms platform and most of them used it in class, we asked for its pros and cons. The advantages mentioned, were: structural elements, familiarity, flexibility, quick and easy assembly, solderless connections, stability of software and hardware, large on-line community, graphical programming, affordable for schools. The disadvantages: the graphical programming gets quickly confusing by increasing project complexity, limitation of the number of sensors (four) and actuators (three) that can be attached to the con-





**Figure 5.1:** Ten secondary school teachers rated the importance of different properties (stability of software, stability of hardware, costs, compatibility with industrial standards or other toolkits, and open-source license) of a robotic toolkit for education based on their personal opinion. The scale ranges between one and seven, where one stands for “not at all” and seven represents “very much”

troller, limited accuracy of actuators and sensors, mechanical clearance of the mechanical construction parts, limitation in the variety of sensors and actuators, “black box” of the hardware, expensive price.

We further asked the teachers to rate the importance of the following properties of an educational toolkit: stability of software, stability of hardware, costs, compatibility with industrial standards or other toolkits, open-source license. Figure 5.1 shows the results. The data needs to be interpreted qualitatively together with the arguments each teacher gave in the interview together with his rating. The stability of the software and the hardware are very important issues for the teachers. Since class time is always limited it has to be assured that no time is being spent by fixing hardware or by restarting systems due to crashed software. Compatibility is less important, the arguments were that once a decision on a platform has been made they would most likely use it as is. The open-source philosophy is welcome, however, schools are often willing to pay for software licenses. We had the most diverging answers on the question concerning costs. For some teachers it is no problem to receive the budget needed, whereas other schools are much more costs sensitive. We learned that this depends highly on the decision maker of the school. In one case, due to a technology savvy principal, any new acquisition had easily been approved. This might explain why the costs are mentioned in both, advantages and disadvantages of the LEGO Mindstorms platform, it just depended on the interviewee.

The teachers were interested to get introduced to the EmbedIT toolkit. Some could consider its use in class once it is released and commercially available. The positive properties pointed out were: the visibility of the electronics (as opposed to the “black box” of LEGO), the possibility to study circuit schematics, open-source license, extendability, competitive price, plug and play property, the fostering of innovative work, variety of sensors, the quick access of sensor data for exploration, possibility to focus on behavior, biological inspired example exercises.

The teachers could see it as an complementary kit to LEGO Mindstorms. Interestingly, the visibility of the electronics was overall appreciated. One teacher said that his explanation to the observed disinterest of young people to understand more about the technology they are immersed in, is because they don’t think they can change anything. They take it as is. If we could show them that they actually can customize it and create their own objects, this might help. Some concerns mentioned about the EmbedIT toolkit were that it might be too fragile for class use (solder connections could break) and consequently they would neither have the time nor the expertise to

fix them. Additionally, the teachers would like to buy a toolkit as is, and not to bother searching for matching components and purchasing them from different suppliers.

## 5.1 Summary

The insights gained by the teacher interviews helped to answer research questions 5–6. We learned regarding classroom practices that every teacher likes to prepare his individual syllabus. Example exercises are very welcome but they are rarely adopted by the teachers as is. They always prefer to adapt them according to their own needs. Teachers share exercises or ideas not very actively (there was only one website mentioned created by one teacher which was frequently accessed by colleagues), resources were shared at seldom occasions (e.g. two schools in the same town taught the EFI classes together, two neighboring schools shared their LEGO Mindstorms kits). We further conclude that schools are highly heterogeneous regarding infrastructure, budget, area of disciplines, characteristics or students, etc. The EFI curricula created by the teachers cover diverse and broad areas in computer science. Some focused on “classical” topics (datatypes, algorithms, logic, etc.) whereas others discussed more “applied” issues, such as security, data privacy, Internet and law.

The survey about the teachers’ requirements for a platform (research question 6) showed that every mentioned aspect is more or less equally important. Through the interviews we understood that a platform is usually used as is. It should therefore be available as one complete kit. The tolerance to fix things (e.g. soldering a broken connection) is not very high among the teachers. The kit should be stable enough to “survive” class environments. But also here there are exceptions. Some teachers considered it as a valuable experience for students to understand why something broke. The visibility of electronic components on a PCB was overall appreciated.

# Classroom Practices and Evaluation of EmbedIT

This chapter covers the practical teaching activities using EmbedIT. We address research questions 7–9 of chapter 1.

- 7. Is it possible to design exercises, which train a diversity of skills using just one platform?*
- 8. How does EmbedIT perform in real classroom environments?*
- 9. What are the best practices to conduct hands-on classes using toolkits?*

Our goal was to develop one platform that enables the creation of a variety of different exercises to explore interdisciplinary fields such as electronics, mechanical engineering, material science, biology, psychology, arts, etc. (research question 7). Group collaboration, time management, creativity, brainstorming and discussion competences are further important skills for successful careers. Section 6.1 describes the exercise examples we designed using EmbedIT.

Research question 8 concerns the performance of EmbedIT in real class environments. Since the beginning we emphasized a practical approach closely related to the target audience. Section 6.2 detail all the classes we carried out using EmbedIT in the context of this thesis.

Designing and conducting hands-on exercises with a large number of students in different schools brings valuable experiences about. This tacit knowledge is gained with increasing experience. Transferring this knowledge would be very helpful for new teachers, however, it is hard to acquire it theoretically. We try to summarize our insights in designing and conducting exercises as best practices in section 6.3 (research question 9).

## 6.1 Exercise Design

We created three main types of exercises each focusing on a specific skill required when building robots and interactive systems. The first exercise type focused on robot control. Here, a more or less fixed robot platform has been provided to the students. The main task was to work on the control of the robot by using the graphical programming interface of the EmbedITApp. The exercise is described in section 6.1.1

The second type of exercise focused on the morphology, i.e. the design of the body of the robot. Here, the control routine was more or less fixed and the students had to find a way to construct the body of the robot such that it performs well. A detailed description about this exercise can be found in section 6.1.2.

The EmbedIT project is affiliated to the AILAB which mainly builds robots for research. Therefore, the platform has originally been designed as a robotic toolkit. Nevertheless, it is not at all constrained to be such. The boundary of what is defined as a robot and what isn't is flexible depending on someone's background and opinion. From an unconventional point of view one would agree that for instance a smart-phone does not differ much from a robot, whereas a classical roboticist would not be comfortable with that idea. One can argue that from a hardware point of view smart-phones are much more sophisticated high-tech devices than personal computers, since they are equipped with a large number of sensors, several networking technologies such as Wi-Fi, Bluetooth and GSM and are actuated (vibration functionality). Even the software of smart-phones is also rapidly evolving to incorporate autonomous and "intelligent" control routines (e.g. the Siri speech recognition software on iPhones, functions that automatically chose the ringtone volume according to ambient sensor input). The smart-phone is already a ubiquitous companion that soon will take over more and more tasks of our everyday lives such as locking doors or making payments. The only difference to classical robots used in AI is that smart-phones do not yet locomote (by themselves).

As a conclusion, the EmbedIT toolkit is suitable for a number of application where sensors and actuators are used within the technical constraints of the toolkit. Based on these thoughts we designed the third exercise which we called the *Physical Computing* scenario. More information about that can be found in section 6.1.3.

### 6.1.1 Focus on robot control

#### Programming state-of-the-art robots

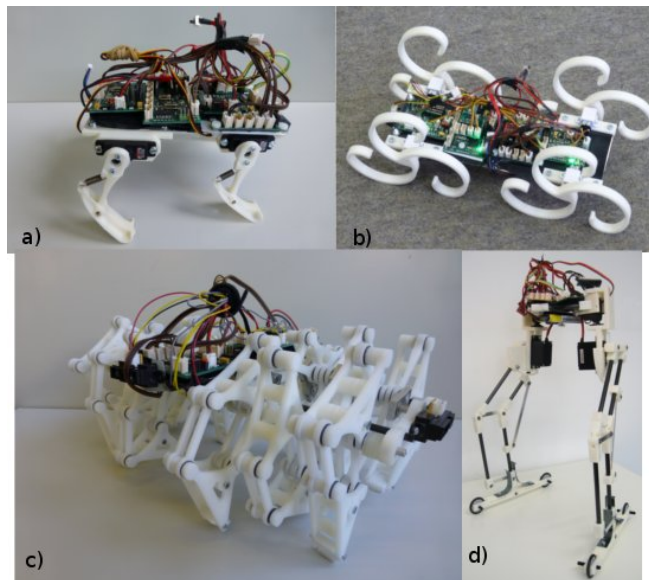
For the control exercise we focused on a robot's control and less on building the body. In any case, the class time would have been too limited to focus on both. Consequently, we decided to provide ready built robots. However, since we came as guests from a research laboratory to the schools for just one special instance, we thought it would be suitable to bring some state-of-the-art robots with us. Further, we wanted to expose the students to some unconventional robots they might never have worked with before. Especially robots that incorporate passive dynamic and compliant properties. Figure 6.1 and 6.3 show the robots we used in class for the control exercises. They are all rebuilt based on robots used in research as well as art projects. The robot in Figure 6.1a) is inspired by the quadruped robot *Puppy* built in our own research laboratory [Iida et al., 2005]. It is equipped with four servo motors, each located at the hip of the passive spring leg. Additionally, we used a smaller version of the *Rhex* robot (Figure 6.1b) [Saranli et al., 2001]. Here the two DC motors on each side of the robot are coupled such that the user can solely control the speed of two motors. Fascinated by Theo Jansen's *strandbeests*<sup>1</sup>, we provided an actuated, smaller version of it (Figure 6.1c). The forth robot was a two legged skating robot built by a master student of our laboratory (Figure 6.1d) [Pils, 2010]. Each robot was connected through Bluetooth to its dedicated Netbook, which had the EmbedITApp running. By means of this application the students could remote control the robots in real time as well as directly define and upload control routines.

#### Braitenberg vehicles

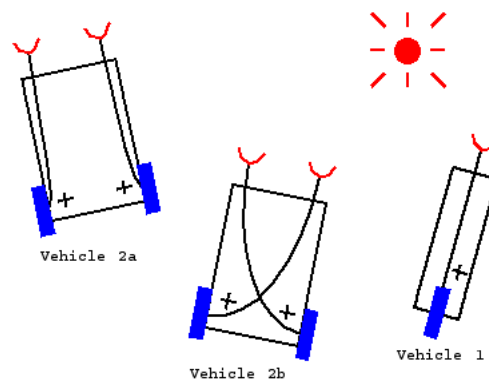
We designed an exercise which is based on the vehicles introduced by neuroscientist Valentino Braitenberg [Braitenberg, 1986]. The students learned about the concept of each vehicle and the connection properties. We used in this exercise vehicles number 1, 2a, 2b, 3a and 3b, as seen in Figure 6.2. The students could verify their thought experiments by testing each vehicle on a real

---

<sup>1</sup><http://www.strandbeest.com>

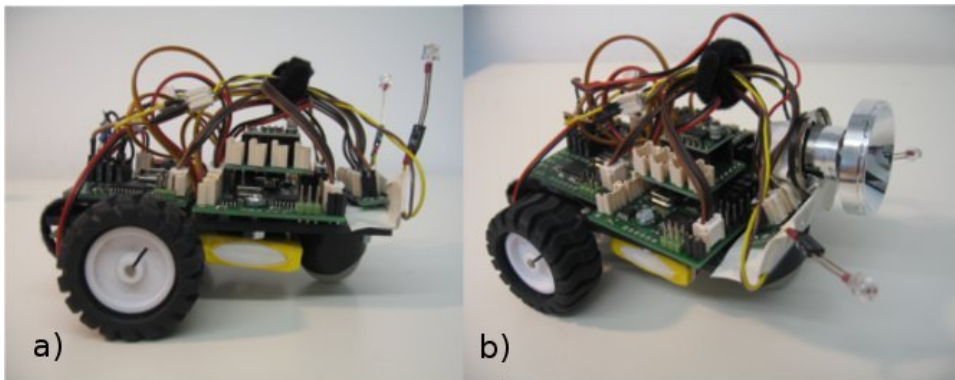


**Figure 6.1:** Robots used in class for the control exercise. a) A quadruped robot actuated by four servo motors. The robot's legs have passive dynamic characteristics. b) A smaller version of the Rhex robot. The two DC motors on each side of the robot are coupled. The wheels have compliant and passive properties. c) An actuated version of the Theo Jansen's strandbeests (this robot has been built by Lijin Aryananda). Only two DC motors control all the legs which are coupled in a sophisticated way. d) A two legged skating robot. Four servos control the angle between the feet as well as the bending of each leg.



**Figure 6.2:** The Braitenberg vehicles used in the context of the MINT workshop at FGZ. The positive symbol means that light increases motor speed linearly. Vehicle 1 maps one sensor value to both motors, 2a and 2b map each sensor directly on either one motor (3a, 3b are the same as 2a and 2b respectively but with an inhibitory sensory motor coupling).

robot. For that we took the robots we built in the “MAS Ergänzungsfach Informatik an Gymnasien” study program with the high school teachers (section 3.1.1, [Assaf and Pfeifer, 2011b]), but this time equipped with the EmbedIT hardware (Figure 6.3a).



**Figure 6.3:** The robot used for the Braitenberg vehicles exercise. a) The two light sensors left and right in the front of the robot are pointing upwards for the initial experiments with the external light source. b) A light source is additionally mounted on the robot for the collective behavior experiment.

## 6.1.2 Focus on robot morphology

We designed an exercise fully dedicated to the exploration of “embodiment”. Traditional robotics curricula focus mainly on the robot control and even if the students have the opportunity to custom build robots, the morphologies are often biased by classical robots. The students are usually not aware of the important role material properties play in interaction with the environment on the performance of a robot. Since EmbedIT is designed to especially enable education in embodied AI, we designed the following novel “soft-bodied robot locomotion” exercise.

The task of robot locomotion is a good case study to show the impact of clever chosen material properties and good robot morphology design on the performance of a robot. Often, solely by changing the robot’s body properties, fast and stable locomotion can be achieved. Students naturally tend to focus on the control program of a robot when trying to increase locomotion stability. Our aim is to disable control programming to force the students to emphasize on a robot’s body. In a number of teaching activities we conducted a LEGO competition in the past, from which we usually received positive feedback [Assaf and Pfeifer, 2011b]. Students had to achieve faster locomotion by simply changing the body of a default quadruped LEGO robot distributed in the beginning of the exercise. A fixed control program was running on both motors of the robot (rotation forwards in constant speed). Inspired by this exercise we created a new one which uses the EmbedIT toolkit as well as soft materials instead of rigid LEGO construction parts. The soft material leads to interesting dynamics and choosing EmbedIT as a robotics platform enables a greater variety of robot morphologies. Students basically had to build soft-bodied robots that locomote in a straight path for a given distance as fast as possible. The control was fixed, but the robot had to be built from scratch.

The tutorial on soft-bodied robot locomotion held in the context of the 2012 ETH Summer School on Soft-Robotics is described in the following article (Appendix D) [Assaf et al., 2012b]:

Dorit Assaf, Xiaoxiang Yu, Liyu Wang, and Fumiya Iida

### *Teaching Soft-Robotics: A Case Study in Building Elastic Creatures that Locomote*

**Abstract** Locomotion has been one of the frequently used case studies in hands-on curricula in robotics education. Students are usually instructed to construct their own wheeled or legged robots from modular robot kits. In the development process of a robot students tend to emphasize on the programming part and consequently, neglect the design of the robot’s body. However, the morphology of a robot (i.e. its body shape

and material properties) plays an important role especially in dynamic tasks such as locomotion. In this paper we introduce a case study of a tutorial on soft-robotics where students were forced to focus solely on the morphology of a robot to achieve stable and fast locomotion. The students should experience the influence material properties exert on the performance of a robot and consequently, extract design principles. This tutorial was held in the context of the 2012 Summer School on Soft Robotics at ETH Zurich, which was one of the world's first courses specialized in the emerging field. We describe the tutorial set-up, the used hardware and software, the students assessment criteria as well as the results. Based on the high creativity and diversity of the robots built by the students, we conclude that the concept of this tutorial has great potentials for both education and research.

### 6.1.3 Focus on physical computing

Physical computing is a term often used in the artist community. It basically means that the physical world is sensed and integrated into the virtual world by means of sensor networks connected to computers. Physical computing as a discipline is usually taught in art and design schools. The curricula cover topics such as programming embedded systems using Arduino, setting up sensor networks using XBee, and using web servers to collect and display the sensor data [O'Sullivan and Igoe, 2004], [Igoe, 2007], [Faludi, 2010]. The aim is to build interactive installations using sensors and actuators. We chose the physical computing scenario as the third exercise type for the evaluation of EmbedIT. We want to demonstrate its openness and flexibility and that it is not limited to be a robotic toolkit. From a hardware point of view the physical computing community and the robotic community don't differ much, however, the applications are very different. We intentionally chose a more "arty" exercise in order to be appealing for those students that are not interested in building robots. The exercises we created are "Rube Goldberg Machine" and "ubiquitous computing".

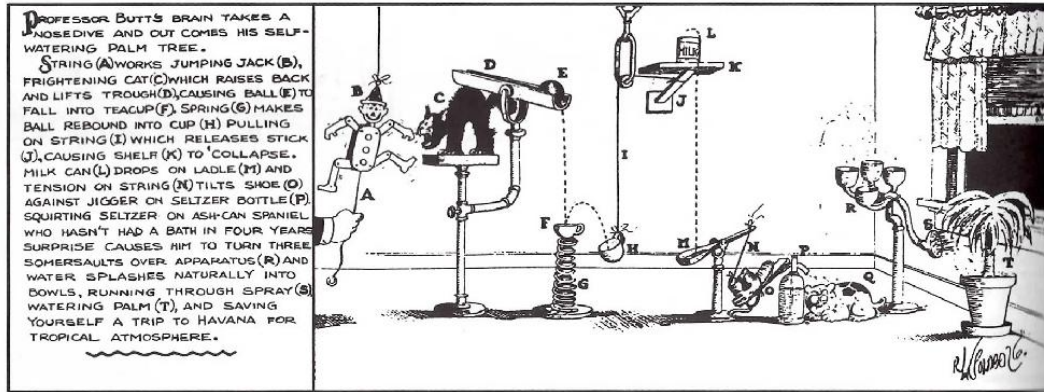
#### Rube Goldberg Machine

A *Rube Goldberg Machine*, named after the American cartoonist and inventor Rube Goldberg (1883-1970), is a deliberately over-engineered machine that performs a very simple task in an unnecessary complex way, often through chain reactions. In his cartoons he drew humorous, unrealistically complicated machines that solve trivial tasks such as turning off the lights, opening the garage door and watering the plant (Figure 6.4). He never built the machines he drew, but his cartoons have become an inspiration to engineers, students and artists across the world that actually constructed similar kind of complicated machines. A large number of art installations exist (e.g. "Der Lauf der Dinge" by Fischli & Weiss in 1987, "Cog" TV commercial by Honda in 2003) as well as yearly Rube Goldberg machine competitions at secondary school level<sup>2</sup>.

The idea to build a Rube Goldberg machine is very appealing and suitable for class. The construction of this machine requires a variety of important skills. The teacher basically just predefines a simple task which has to be executed by the machine. The students have to be very creative to come up with fun ideas for the chain reactions. The students should not be constrained by a fixed mechanical construction kit, moreover they should be encouraged to use any object they can find around them. The more a daily-life object is used in an unusual way to release the next step in the machine's chain reaction, the more interesting and fun it is. Further, the chain reactions are not more than simple sensing and actuation actions. The students learn about different ways how a chain reaction can be implemented (either through purely mechanical "sensing" and actuation such as for instance a ball rolls and pushes another object, or through sensors that are coupled with actuators using a predefined control program). As a next step the students have to find mechanical construction solutions to incorporate the sensors and actuators in the machine,

<sup>2</sup><http://rubegoldberg.com/?page=contest>

### Self-Watering Palm Tree



**Figure 6.4:** A cartoon of Rube Goldberg. An unnecessarily complex machine to water a palm tree. Picture courtesy of <http://www.rubegoldberg.com>

sensors have to be calibrated, control routines have to be programmed. Additionally, the very important skills needed in this exercise are team work and time management. Opposed to typical robot education activities, where teams compete against each other, here the students have to work together to create one common artwork. Every team is responsible for an equally important subpart of the machine (i.e. one or several chain reactions). The students have to collaborate with their neighboring teams on how to interface their individual chain reactions together. Each group's work is equally important and crucial for the success of the machine. Time management plays an important role as well if the machine has to be finished on time. Especially because the students usually underestimate the consumed time for testing, since no Rube Goldberg machine runs neatly in the first shot. The final result is an artwork with equal contribution by every team. The machine can further be filmed and put on-line for friends and family.

This exercise concept might be especially appealing for students that normally don't like competitions. Further, since many competitions (especially robotic competitions) usually value just one simple property or behavior of an agent (speed, navigation), students consequently stick to established effective solutions (e.g. building a wheeled robot to achieve fast locomotion) rather than exploring more creative and unusual ways to solve a problem. With this Rube Goldberg machine exercise out of the box thinking is appreciated and fostered.

### Ubiquitous Computing

We designed the ubiquitous computing exercise to emphasize the pervasive use of computers, sensors and actuators in modern societies. Young people today grow up being immersed in technology. They are said to be savvy in using computers, smart phones, digital cameras etc. As seen in the previous section 2.3, the technology young people pervasively use is taken for granted and not really considered "technology". The students are proficient in mastering the technology but have only little knowledge about the technical details. With this exercise we tried to establish the missing link between the student's daily life technology and the components its made of. Further, we tried to foster creative thinking with a short brainstorming session.



## 6.2 Classes and Workshops

We emphasized testing as an concurrent process during the development of EmbedIT. We tested all components of the toolkit at an early stage of the design process. Electronic boards and their corresponding firmware have been integrated in robots built in the context of other research projects at the AILAB (e.g. MESTRAN leg [Hung et al., 2012]). This helped to find firmware bugs as well as flaws and weaknesses of the electronic circuit design. The graphical user interface (EmbedITApp) has further been tested with high school students and formally evaluated in the context of a masters thesis [Sterchi, 2011]. The proposed GUI design change requests which based on state-of-the-art GUI design principles as well as on the evaluation results of the high school students have been implemented in newer versions of the EmbedITApp software.

The *Ergänzungsfach Informatik* (EFI) is usually taught weekly during two or three lessons of approximately forty minutes each within the last two years before the *Matura* (this certificate eligibles enrollment for any subject in University). More general information about the EFI classes can be found in the teacher interview section 5. The teachers could free up one weekly session for our evaluation of the toolkit (in Table 6.1 referred to as “EFI class”). Occasionally, schools and Universities dedicate a special workshop week to a specific topic in order to motivate and encourage students to pursue a career in these related disciplines. Some teachers could provide us with more time during these special school theme weeks (in Table 6.1 referred to as “EFI day or EFI week”).

During the five months evaluation period between May 5<sup>th</sup> 2012 and October 4<sup>th</sup> 2012 all three exercises had been conducted in the context of EFI classes or special workshop weeks, as seen in Table 6.1. The exercises were distributed among the nine participating schools as follows: control exercise was held in four schools (54 students), the morphology exercise in three (89 students), and the physical computing exercise in three (36 students). In total 179 students did an exercise using the EmbedIT toolkit (167 different people worked with the toolkit). The following sections 6.2.1-6.2.3 describe the exercise results in detail.

### 6.2.1 Control exercise results

To introduce the robots we showed videos of the original robots and how they are used in research. We collected feedback through questionnaires as well as personal conversations. Older students were aware that the GUI of the EmbedITApp is still under development, so they were happy to make suggestions for improvements as well as to document software bugs. The questionnaire consisted of questions concerning the students background, their previous knowledge of robot toolkits and programming languages, and their estimated difficulty to build and control the research robots we demonstrated in the videos. Further, we wanted to know how much they liked the exercise, how user friendly the GUI was, and whether they would like to work with the toolkit again.

We conducted these robot control exercises in two instances of an EFI class and in two occasions in the context of a workshop within a special theme week of a school. Totally, 54 students between 12 and 19 years did the exercise. The detailed evaluation results for the EFI classes and the workshops during events and special theme weeks are described in the following sections.

#### EFI class Gymnasium Rychenberg, May 10<sup>th</sup> 2012

The Gymnasium Rychenberg is located in Winterthur which is the second biggest city in Kanton of Zurich, Switzerland. The EFI class of the Rychenberg school is held together with a neighboring secondary school, the Büelrain. Students from Rychenberg major in modern languages whereas the focus of the Büelrain school is economics. Nine students attended the class, six of which

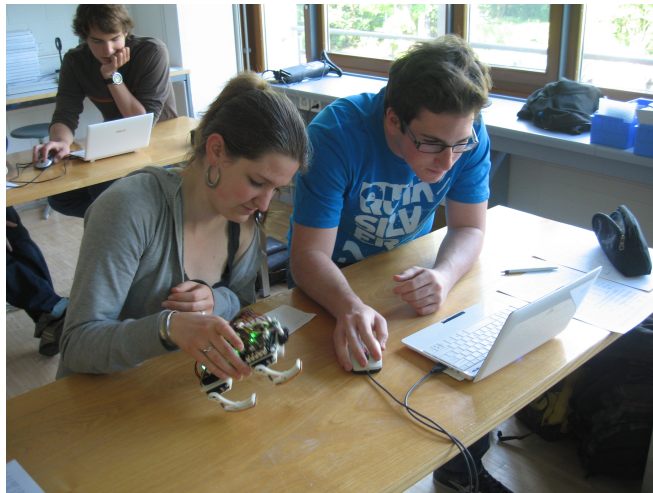
**Table 6.1:** Overview of all classes and workshops conducted in the context of the EmbedIT evaluation

School - Event	Date	Control exercise	Morphology exercise	Physical computing exercise
EPFL - Robotic Festival	05.05.2012	20 students (12-18 years)		
Gymnasium Rychenberg - EFI class	10.05.2012	9 students (18-19 years)		
Gymnasium Immensee - EFI class	27.06.2012	11 students (17-18 years)		
Freies Gymnasium Zürich - MINT days	04.10.2012	14 students (14-17 years)		
ETH - Summer School on Soft-Robotics	20.06.2012		59 students (24-36 years)	
Kantonsschule Seetal - special EFI day	25.09.2012		18 students (16-19 years)	
Kantonsschule Trogen - special EFI week	26.09.2012 & 27.09.2012		12 students (17-19 years)	12 students (17-19 years)
Zürcher Hochschule der Künste - Junior Design Department	17.07.2012			12 students (12-16 years)
Kollegium Spiritus Sanctus Brig - EFI class	29.08.2012			12 students (17-20 years)
<b>Total 179 students</b>		<b>54 students</b>	<b>89 students</b>	<b>36 students</b>

attended the Rychenberg school and three the Büelrain. The students were in the last month before graduation, therefore between 18 and 19 years old (average 18 years, among them one female student). To the question about their future study plans, the ones from the economics major answered computer science or business informatics, the others that hadn't made up their mind yet mentioned some disciplines such as medical science, mechanical engineering or geology.

These students did the GUI test run nine months before [Sterchi, 2011] (see section 6). So they were familiar with the software in an earlier state of development. The students appreciated the progress of the user interface and that some of their recommended improvements have been implemented. With the exception of one student, all have worked with robots before, namely the LEGO Mindstorms platform. Eight out of nine students had Java programming knowledge, three more were additionally familiar with PHP and HTML.

In these eighty minutes class time, groups of two students were formed and two Puppys, one Skaty, one Strandbeest and one Rhex robot were distributed. Rhex and the Strandbeest were additionally equipped with a number of infrared distance sensors. The students programmed the robots using the graphical programming environment of the EmbedITApp. Students working with Skaty or Puppy focused on tuning the servo motor sine parameters to achieve stable locomotion (Figure 6.5). The groups with Rhex and Strandbeest implemented mostly obstacle avoidance behavior. At the end of the class we asked for their favorite robot among the presented ones. 25% liked Puppy, 50% Rhex, 12.5% Strandbeest and 12.5% Skaty. It has to be mentioned that some students had the opportunity to work with two or more different robots, whereas some students spent the whole class time on Puppy. So some students argued that they liked a specific robot more is because they did not try any others. Others liked the fact that Rhex is fast and very simple in its construction. Strandbeest was credited for its nice looks and way of locomotion.



**Figure 6.5:** Students programming locomotion gaits on the Puppy robot in the context of an EFI class at Gymnasium Rychenberg.

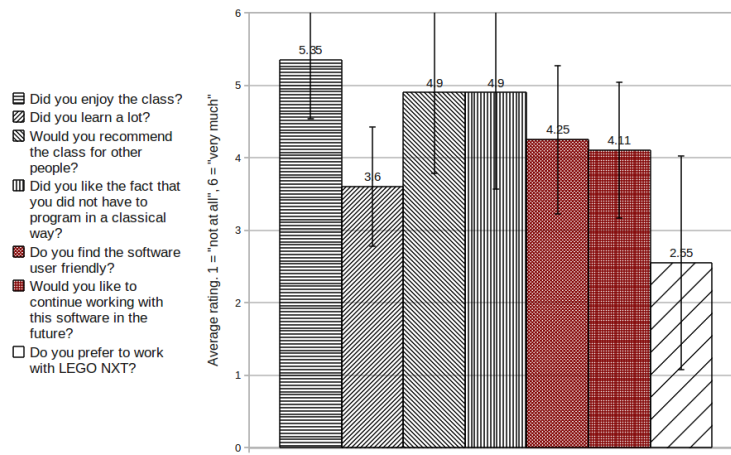
We asked them how difficult it was to control each robot they tried. From a scale from one to six, where one represents “very easy” and six “very difficult”, Rhex was estimated around two, Strandbeest around three and Puppy around five. Unfortunately one motor of Skaty broke during class time such that only one group had the chance to work with it.

### EFI class Gymnasium Immensee, June 27<sup>th</sup> 2012

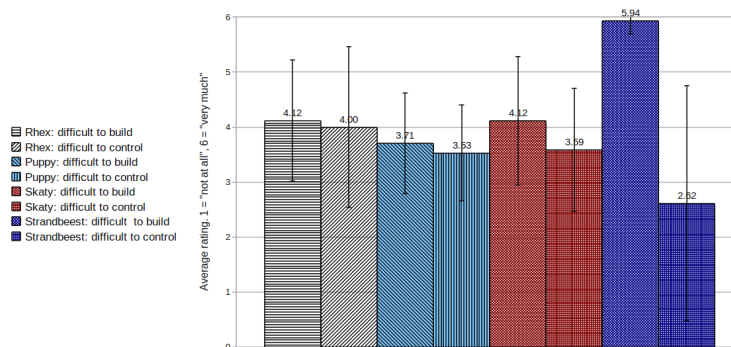
Immensee is a private school in Kanton Schwyz, Switzerland. Eleven students between 17-18 years old (average 17 years), that major in natural sciences attended the hundred minutes EFI class (45% female students). All students were experienced with LEGO Mindstorms as well as Java programming. Many answered medical sciences, physics and some other natural science disciplines as their future study plans. It has to be mentioned that graduation is still two years away for those students, so these preferences can surely change. Anyhow, this information is still meaningful to get an idea about the students background and preferences.

We distributed three Puppy, two Rhex and one Strandbeest robots. To make sure that this time all of the students were able to work at least with two types of robots, we switched robots after half of the class time. In this class the Strandbeest and Puppy were the favorite robots with each 45% of the votes. Puppy was liked for its bio-inspiration as well as for the fact that it was more challenging to achieve a stable gait. The Strandbeest was credited for its sophisticated design and its elegant movements. For the control difficulty estimation, Rhex and Strandbeest were graded more easy with a three and Puppy a bit more difficult with a four.

**Survey results** The following survey results combine the two EFI classes at Gymnasium Rychenberg and Gymnasium Immensee together. Figure 6.6 shows the students answers about general questions such as “Did you enjoy the class?”, “Do you find the software user friendly?”, or “Do you prefer to work with LEGO NXT?”. The students had to rate each question between one to six, where one stands for “not at all” and six for “very much”. We appreciated the fact that the students enjoyed the class and that they found the software more or less user friendly. The questions about whether they learned a lot was rated with a 3.6 which can be explained by the limited class time available for in depth background information. Nevertheless, there is room for improvement here.

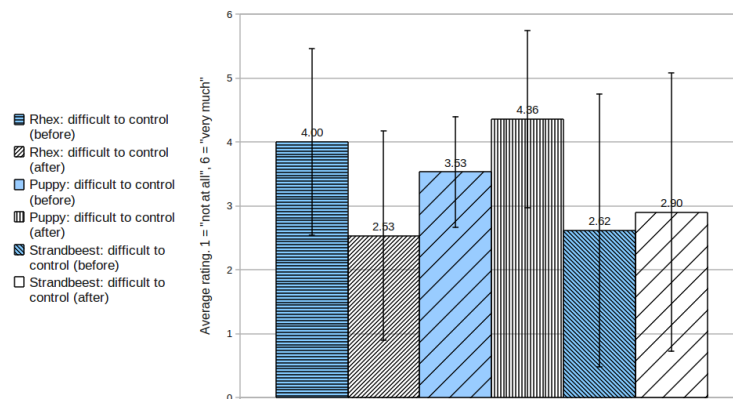


**Figure 6.6:** Combined survey results of the EFI classes at Gymnasium Rychenberg and Gymnasium Immensee (total 20 students). The students had to rate general questions about the class between one and six, where one stands for “not at all” and six represents “very much”.



**Figure 6.7:** Combined survey results of the EFI classes at Gymnasium Rychenberg and Gymnasium Immensee (total 20 students). The students had to estimate the difficulty to build and control the Rhex, Puppy, Skaty and Strandbeest robots based on video footage. The scale ranges between one and six, where one stands for “not at all” and six represents “very much”.

At the beginning of the class we showed videos of the original robots and asked them for their estimate about the difficulty to both build and program each robot. With this question we wanted to get an idea what they think about programming and constructing robots. Figure 6.7 shows their estimation before they worked with the robots. After the class we asked for the difficulty in programming based on their experiences, which is summarized in Figure 6.8. The difficulty to control Puppy was initially underestimated. It gained drastically in the difficulty raking in relation to Rhex.



**Figure 6.8:** Combined survey results of the EFL classes at Gymnasium Rychenberg and Gymnasium Immensee (total 20 students). The students had to rank the difficulty to control the Rhex, Puppy and Strandbeest robots based on their experiences in class. The actual difficulties are plotted next to their initially estimated difficulties. The scale ranges between one and six, where one stands for "not at all" and six represents "very much".

### Workshop at Robotic Festival EPFL, May 5<sup>th</sup> 2012

The École Polytechnique Fédérale de Lausanne (EPFL) Robotic Festival<sup>3</sup> was held for the 5<sup>th</sup> time at its campus in Lausanne. It is targeted to families to come and experience robotics research. Next to a big robot exposition, a large number of hands-on workshops are usually offered for kids of all ages. Due to its popularity the Robotic Festival grew rapidly in the past years and reached up to 15'000 visitors in 2012. In the 21 workshops that were offered in 2012 students could build little electronic devices, could assemble small robots and learn how to program (in different levels of expertise). Our workshop was called "piloteur des robots marcheurs" or "controlling walking robots", as described in Figure 6.9. The kids had the opportunity to control the Rhex, Puppy, Strandbeest and Skaty robots using the graphical user interface in a three hour class (Figure 6.10). The same class was held once in the morning and once in the afternoon with each ten students between the age of 12 and 18 (the average age was 14). From the total twenty students 20% were female. About half of the students had worked with the LEGO Mindstorms or Thymio platform before. 30% of the students had visited the Robotic Festival in previous years. This workshop was the first time we tested the Rhex, Puppy and Skaty robots in a class environment (Strandbeest has been used at the AILAB before in a class for undergraduates). Concerned whether the 3D printed parts of the robots will hold through the entire class time we brought a lot of spare parts with us. Interestingly, none of the 3D printed parts broke, in contrast to the motors which turned out to be the weak spot of the robots. Half of all servo motors and DC motors had to be replaced during the day due to jammed gears. However, it has to be mentioned that the kids enjoyed speed and often run the motors at their maximum capacity.

We decided not to integrate the collected questionnaire data from this workshop. The environment was somewhat informal and therefore it was difficult to be sure the acquired data is valid. The workshop was in the middle of the general exhibition and frequently visited by a large number of people. Some students left in the middle of the workshop and the empty workplace was instantly occupied by new students which decided spontaneously to jump in. All of a sudden also some parents joined and played with the robots. We enjoyed much the general interest and the dynamic of the workshop but decided not to further process the collected data.

<sup>3</sup><http://festivalrobotique.epfl.ch>



**FESTIVAL ROBOTIQUE**  
SAMEDI 5 MAI 2012 • EPFL LAUSANNE

robotics  
EPFL  
ÉCOLE POLYTECHNIQUE  
FÉDÉRALE DE LAUSANNE

**ACCUEIL**  
**PROGRAMME**  
**EXPOSITIONS**  
**ATELIERS**  
**SPECTACLES**  
**PICOBOTS**  
**ENCADRANTS**  
**CONTACTS**  
**ACCÈS**

page en version imprimable  
rejoignez nous sur facebook

► **No 15 : Piloter des robots marcheurs**  
Âge conseillé: de 14 à 19 ans, gratuit, salle CM 113

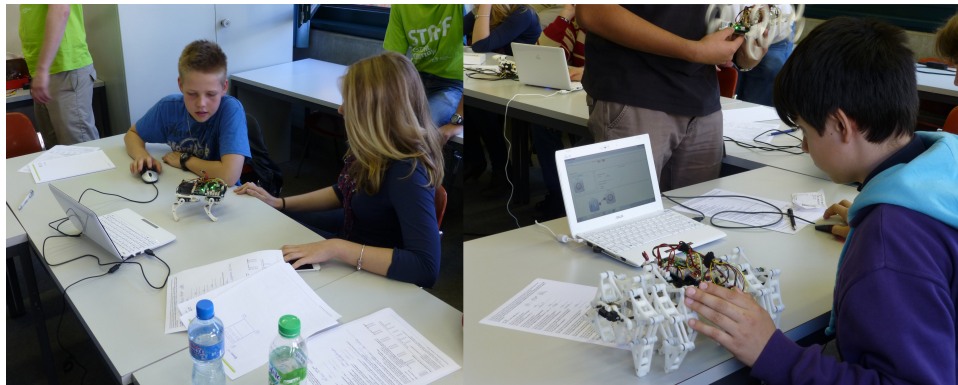
Cet atelier sera bilingue français-allemand

Cet atelier accueille des enseignants à titre d'encadrant. Pour en savoir plus, prière de s'inscrire sous la rubrique encadrants.

**Descriptif:**  
Plusieurs robots différents attendent que vous leur appreniez à marcher! Vous pourrez jouer avec des robots à quatre ou douze pattes, d'autres qui rampent ou alors organiser un "patinage" de robots. Cet atelier ne demande pas de programmer les robots: vous pouvez les laisser courir et régler leurs paramètres pour leur conférer une locomotion stable, élégante ou rapide. Un petit concours sélectionne le robot gagnant!

Organisé par :  
Dorit Assaf, Université de Zurich (Artificial Intelligence Laboratory)

**Figure 6.9:** The workshop description "piloter des robots marcheurs", or "controlling walking robots" at the EPFL Robotic Festival 2012.



**Figure 6.10:** Students programming the Puppy and Strandbeest robots at the EPFL Robotic Festival 2012.

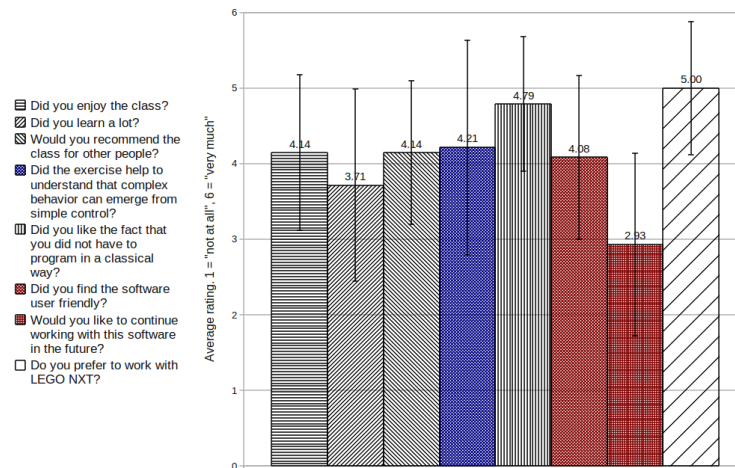
## Workshop at MINT days FGZ October 4<sup>th</sup> 2012

The Freies Gymnasium in Zürich is a private school in the city of Zurich. In the context of the special MINT (math, informatics, natural sciences, technology) days we conducted the Braitenberg vehicles exercise. 14 students between 14 and 17 years (average 15) attended this three hour exercise (among them 71% female students). 50% major in natural sciences, 36% in economics, and 14% in modern languages. Only two students had prior programming knowledge.

Six robots and light sources (torches) were distributed and the teams could initially test each vehicle's behavior. Afterwards, the light source was mounted on the robot (see Figure 6.3b) to test the emergent collective behavior between all robots. Unfortunately, the collective experiment did not work so well due to technical problems of three robots at the end of the class.

Figure 6.11 shows the survey results of this class. The technical problems and the consequent waiting time for the teams with non functional robots might have influenced the rating of this exercise. Defining the sensory motor relationships on the graphical user interface was found not to be very intuitive. Nevertheless, in the free comment section of the questionnaire 50% of the students mentioned explicitly positive the fact that they could verify the thought experiments by means of a real experiment. Most of the negative comments were the software responsiveness and





**Figure 6.11:** The survey result of the Braitenberg vehicles class (total 14 students). The scale ranges between one and six, where one stands for "not at all" and six represents "very much".

the non-intuitive user interface (which is also reflected in the corresponding bar charts in Figure 6.11). The class was introduced to LEGO Mindstorms the day before in the context of another workshop. This could also explain the high positive rating for LEGO Mindstorms in these survey results, due to its novelty factor.

## 6.2.2 Morphology exercise results

This exercise was conducted three times during the evaluation phase of EmbedIT. Once for undergraduates and postdocs in the context of the 2012 ETH Summer School on Soft-Robotics. The other two exercises were given to secondary school students. The results of the summer school are summarized in [Assaf et al., 2012b]. The results of the classes in secondary schools (Kantonsschule Trogen and Kantonsschule Seetal) are described in the following sections.

### ETH Summer School on Soft-Robotics, June 20th 2012

The tutorial on soft-bodied robot locomotion held in the context of the 2012 ETH Summer School on Soft-Robotics is described in the following article (Appendix D) [Assaf et al., 2012b]:

Dorit Assaf, Xiaoxiang Yu, Liyu Wang, and Fumiya Iida

#### *Teaching Soft-Robotics: A Case Study in Building Elastic Creatures that Locomote*

**Abstract** Locomotion has been one of the frequently used case studies in hands-on curricula in robotics education. Students are usually instructed to construct their own wheeled or legged robots from modular robot kits. In the development process of a robot students tend to emphasize on the programming part and consequently, neglect the design of the robot's body. However, the morphology of a robot (i.e. its body shape and material properties) plays an important role especially in dynamic tasks such as locomotion. In this paper we introduce a case study of a tutorial on soft-robotics where students were forced to focus solely on the morphology of a robot to achieve stable and fast locomotion. The students should experience the influence material properties exert on the performance of a robot and consequently, extract design principles. This tutorial was held in the context of the 2012 Summer School on Soft Robotics at ETH Zurich, which was

*one of the world's first courses specialized in the emerging field. We describe the tutorial set-up, the used hardware and software, the students assessment criteria as well as the results. Based on the high creativity and diversity of the robots built by the students, we conclude that the concept of this tutorial has great potentials for both education and research.*

### **Kantonsschule Trogen, September 27<sup>th</sup> 2012**

Trogen is a small village in the rural area of eastern Switzerland. In the context of a special EFI week the students had the opportunity to visit our laboratory. Additional to a lecture and a labtour the 12 students between 17-19 years old (average 18, two female students) attended this three hour workshop (Figure 6.12). 83% major in natural sciences, the rest in economics and arts. All students had experiences with the LEGO Mindstorms platform as well as programming knowledge (mostly Java and C++). As future study interests 67% mention technical disciplines, followed by other disciplines such as economics, medical doctor, or pedagogy. From the five competing teams, only two robots produced valid results (all the others never made it to the goal line). The fastest robot traveled 11.6 seconds to the goal.

The survey results are combined with the ones of the Kantonsschule Seetal in Figure 6.13.

### **Workshop EFI day Kantonsschule Seetal September 25<sup>th</sup> 2012**

The school is located in Baldegg, which is a rural village in the central part of Switzerland. In the context of a special EFI day, the 18 students between 16-19 years old (average 17, two female students) visited our laboratory and additionally, attended this workshop. The students' schedule was tight, so only two hours had been available for the workshop, which is actually too short. From the nine competing teams solely two robots produced a valid measurement. The fastest among them traveled 10.6 seconds to the goal. The reason for the small number of successful robots was the limited time. Many teams were still constructing the robots when the class was over. 55% of the students mentioned that they wished the class would have lasted longer. Nevertheless, the constructed robots were very interesting and the class was found to be a lot of fun.

A collection of robots built by both schools, Kantonsschule Trogen and Seetal can be seen in Figure 6.14.

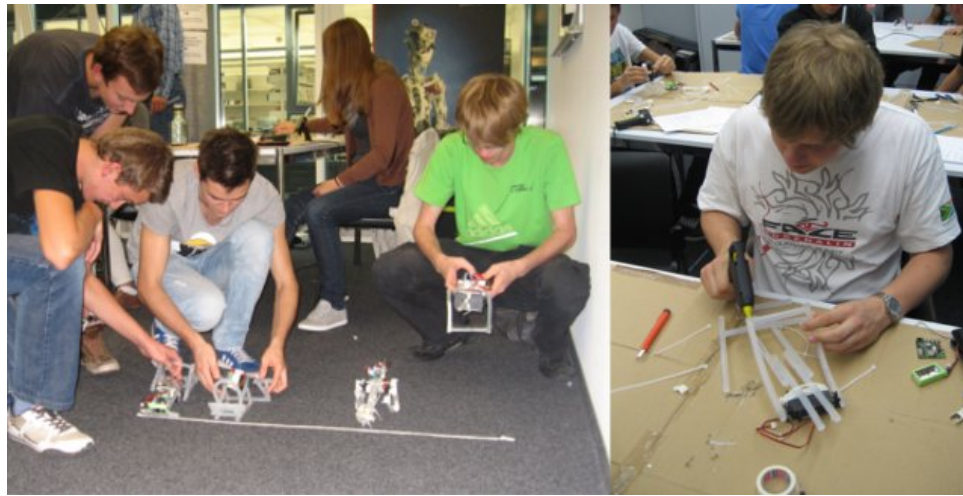
## **6.2.3 Physical computing exercise results**

We conducted the Rube Goldberg Machine exercise in two occasions, at the Zürcher Hochschule der Künste (ZHdk) and Kantonsschule Trogen with total 25 students. The ubiquitous computing exercise was carried out once with 12 students at the Kollegium Spiritus Sanctus Brig.

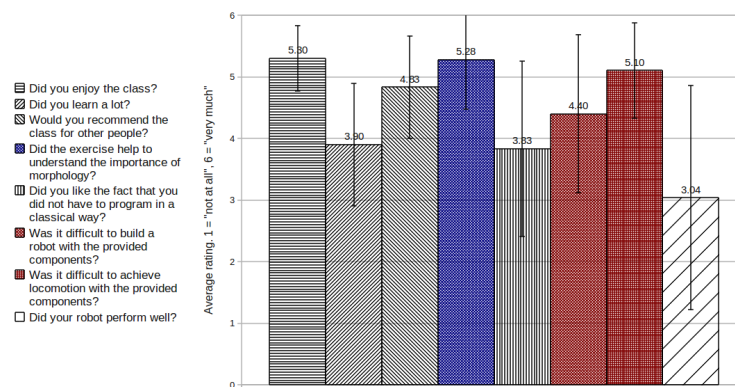
### **Junior design department at ZHdk, July 17<sup>th</sup> 2012**

The ZHdk interaction design department gives kids the opportunity to learn about design in a yearly workshop week called "Junior Design Department". The goal was to build a Rube Goldberg machine in one day (6 hours) using the EmbedIT toolkit. Twelve students between 12 and 16 years old (average age 14, two female) attended this workshop. We formed five teams and provided a number of different sensors for exploration: Force sensitive resistors (FSR), accelerometers, gyroscopes, switches, flex sensors, liquid level sensors, tilt sensors, piezo vibration sensors, different kinds of potentiometers, compass, temperature sensors (digital and infrared), pressure sensors, different distance sensors, light sensors. The provided actuators were vibration motors, servo and DC motors. After the exploration phase the students chose the sensors and actuators for





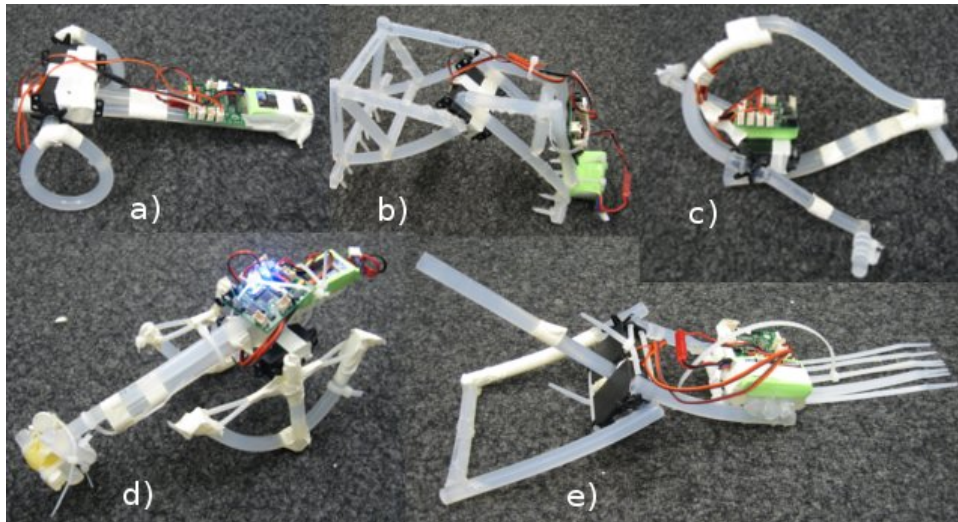
**Figure 6.12:** Students of Kantonsschule Trogen and Seetal are building soft-bodied robots for locomotion.



**Figure 6.13:** Combined survey results of the soft-bodied robot locomotion exercise of both Kantonsschule Trogen and Seetal (total 30 students). The scale ranges between one and six, where one stands for "not at all" and six represents "very much".

their part of the chain reaction. We further provided mechanical construction parts from LEGO, Meccano and Stokys as well as a other material such as cardboard, duct tape etc. (Figure 6.16c and d).

The Rube Goldberg machine consisted of five subparts. Part one: A small ball is released which pushes down a flap over a light sensor. The decreasing light level is detected and releases the LEGO car (Figure 6.15a). Part two: The bumper of the LEGO car bends a flex sensor located at the edge of the table which turns the white wheel (Figure 6.15b). The white wheel pushes the small car wheel which knocks over the lit candle. Part three: the temperature sensor detects the candle and turns a servo motor. Part four: The servo motor rolls a horizontal bottle which pours water into a container. The liquid level sensor detects the increased liquid level and moves a servo motor (Figure 6.15c). Part five: the servo motor pushes a ping-pong ball which rolls down a ramp. At the end of the ramp the passing by ball is detected by a distance sensor located at the rear of the waiting car. The car is released and pushes a pointed stick to a balloon filled with confetti. The balloon bursts and the confetti is released (Figure 6.15d).



**Figure 6.14:** A collection of soft-bodied robots built by students of the Kantonsschule Trogen and Seetal. b) and c) were among the fastest robots.

Of course the whole chain reaction was somewhat ambitious and did not work in the first time. We needed several tries until it finally worked. The execution duration was seven seconds at a length of six meters.

The students feedback was overall positive. The survey results can be seen in Figure 6.17.

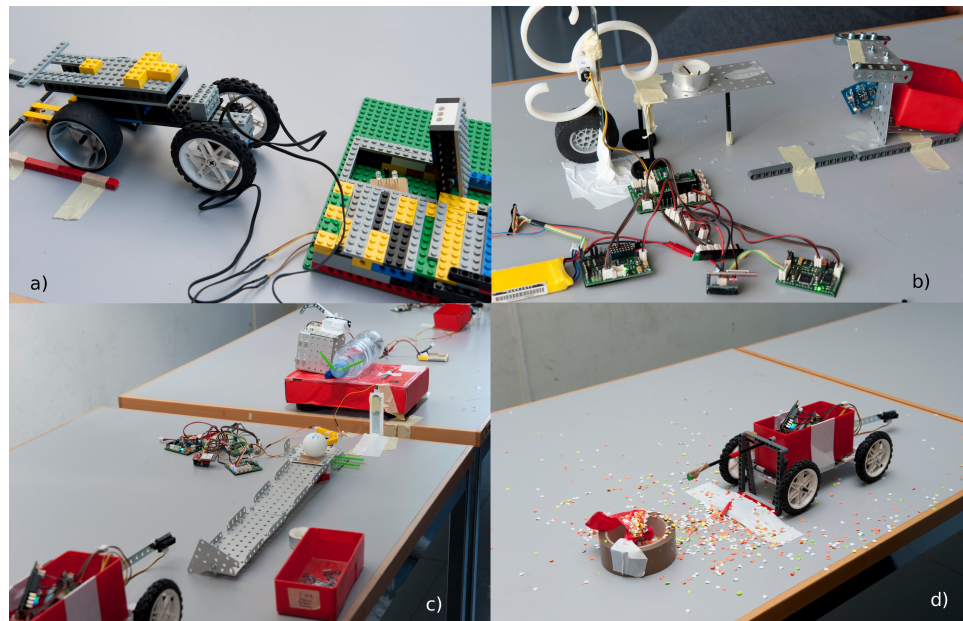
### Kantonsschule Trogen, September 26<sup>th</sup> 2012

The same students from Kantonsschule Trogen that attended the soft-bodied robot locomotion exercise mentioned in section 6.2.2 built a Rube Goldberg machine the previous day (Figure 6.18). The full day workshop duration was still a little too short for building such a machine. Only half of the machine was ready and worked (though some parts were also quite ambitious such as a marble that was supposed to be tossed by a catapult into a cardboard box). Many students mentioned explicitly in the free comment section of the questionnaire that they enjoyed the creative work by constructing the parts of the machine. The survey results can be seen in Figure 6.19.

### Kollegium Spiritus Sanctus Brig, August 29<sup>th</sup> 2012

Brig in Kanton Wallis is located in the Swiss alps close to the Italian border. Almost 1000 students are enrolled in the The Kollegium Spiritus Sanctus. We had the opportunity to give the exercise during a weekly 90 minutes EFI class time. 12 male students between 17-20 years (average 18) attended the class. Five of the students had worked with the LEGO platform before and all of them had Java programming knowledge. 66% of the students major in natural sciences the rest in Economics or modern languages.

We provided a number of different sensors (force sensitive resistors (FSR), accelerometers, gyroscopes, switches, flex sensors, liquid level sensors, tilt sensors, piezo vibration sensors, different kinds of potentiometers, compass, temperature sensors (digital and infrared), pressure sensors, different distance sensors, light sensors) and actuators (servo motor, DC motor, vibration motor). The sensors and actuators were distributed among six stations. Groups of two students were formed and each group had 10 minutes to test the sensors and actuators at each station. After 10



**Figure 6.15:** The constructed Rube Goldberg machine. a) A small ball is released which pushes down a flap over a light sensor. The decreasing light level is detected and releases the LEGO car. The bumper of the LEGO car bends a flex sensor which turns the white wheel. b) The white wheel pushes the small car wheel which knocks over the lit candle. The temperature sensor detects the approached candle and turns a servo motor. c) The servo motor rolls a horizontal bottle which pours water into a container. The liquid level sensor detects the increased liquid level and moves a servo motor. The servo motor pushes a ping-pong ball which rolls down a ramp. At the end of the ramp the ball is detected by a distance sensor located at the rear of the waiting car. d) The car is released and pushes a pointed stick to a balloon filled with confetti. The balloon bursts and the confetti is released.

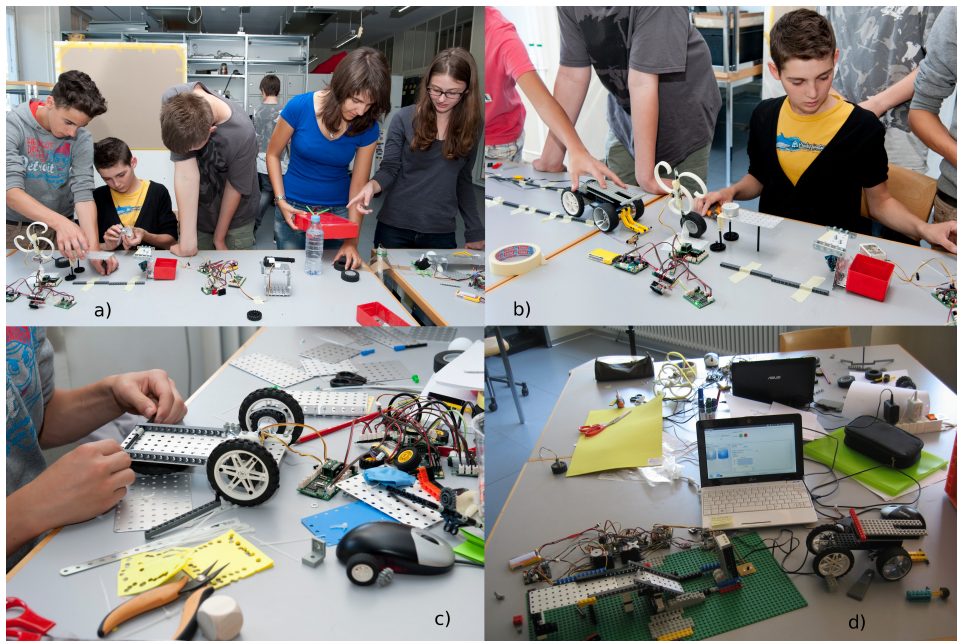
minutes the groups rotated across the stations, such that each group got to play with all sensors and actuators. The sensor values were read out in real time by means of the graphical user interface of EmbedIT. Additional information about each sensor (how it works, what it is used for) was given at each station, with the aim to establish the missing link between their daily life technology and the technical backgrounds. We explained for instance that the gyroscope, accelerometer and compass are usually used in combination for 3D tracking, which is used in Nintendo Wii or smart phones. We explained that smart phones are equipped with built-in vibration motors, like the one they are playing with.

After this we had a short brainstorming session. We wanted to know what ubiquitous computing projects the students could imagine to build using these sensors and actuators. We showed some example videos of projects found on Youtube. Among them an automatic cat feeder: an installation where the cat wears a RFID tag and a feeding station that opens the respective feeding dish (each cat has its own). Another video showed an automatic plant watering system: a sensor measured the humidity of a plant's soil and a pumping system waters the plant if needed.

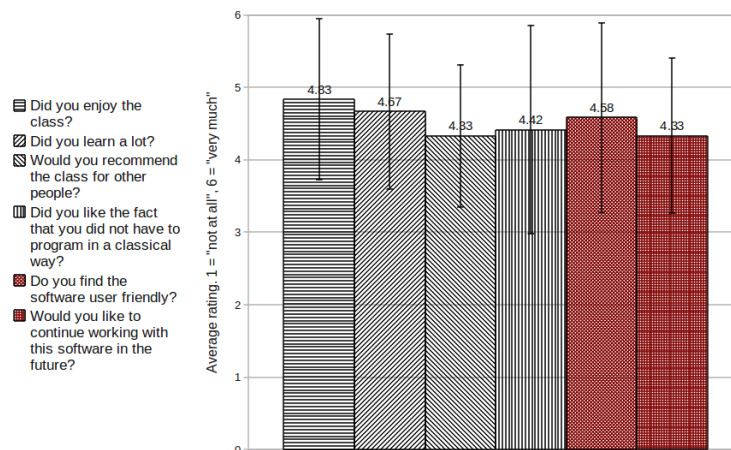
We pointed out that any crazy or maybe impossible idea is welcome for this brainstorming session. Interestingly, the students mainly came up with traditional ideas such as automatic barriers, parking aids, automatic lighting, or automatic counting of people entering a room. Nevertheless, some new ideas were mentioned like an automated regulator for a hot-air balloon (barometer and air temperature that open or close the valve), an automatic bartender or a robotic glove.

Figure 6.20 shows the results of the distributed questionnaire. In personal conversation with the teacher we concluded that the exercise was interesting for the students but the 90 minutes





**Figure 6.16:** Students working on a Rube Goldberg machine. a) and b) setting up chain reaction number two (flex sensor turns wheel which releases a small wheel), three (small wheel knocks over a candle which is detected by the temperature sensor and moves servo) and four (servo pushes a bottle which pours water into a container. A liquid level sensor detects the increases liquid level and moves another servo motor). c) and d) Students combined mechanical construction kits such as LEGO, Meccano and Stokys with other material such as cardboard and duct tape.

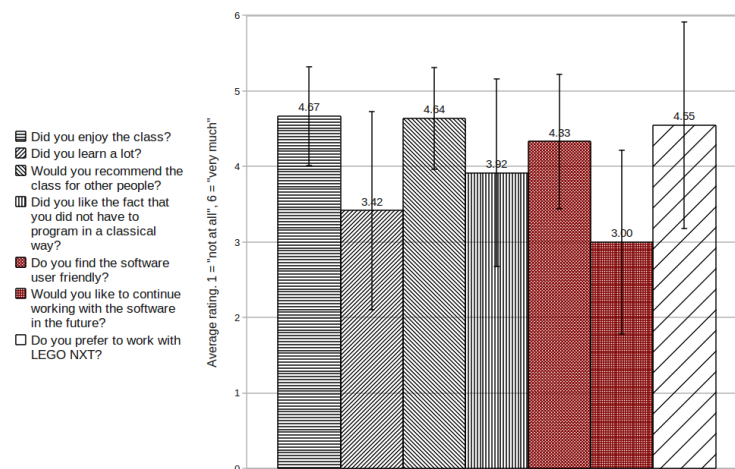


**Figure 6.17:** Survey results of the Rube Goldberg machine exercise at the workshop day of ZHdK "Junior Design Department" (total 12 students). The bar chart shows the average rating to different questions concerning the class. The scale ranges between one and six, where one stands for "not at all" and six represents "very much".

were too short. The physics of the sensors should be explained thoroughly. The teacher pointed out that he liked the fact that the students were exposed to the sensors in their "raw" form. Usually

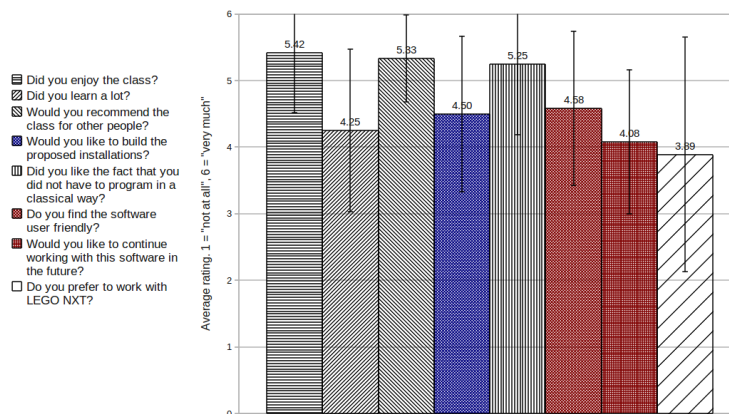


**Figure 6.18:** Students of the Kantonsschule Trogen are building a Rube Goldberg machine.



**Figure 6.19:** Survey results of the Rube Goldberg exercise at the Kantonsschule Trogen (total 12 students). The bar chart shows the average rating to different questions concerning the class. The scale ranges between one and six, where one stands for "not at all" and six represents "very much".

they are hidden within the devices and the students were surprised when they saw the actual sensors and how they look like.



**Figure 6.20:** Survey results of the ubiquitous computing exercise at the EFI class of Kollegium Spiritus Sanctus Brig (total 12 students). The bar chart shows the average rating to different questions concerning the class. The scale ranges between one and six, where one stands for “not at all” and six represents “very much”.

## 6.3 Best Practices for Hands-on Classes

There are many critical factors to consider when designing exercises. The inhomogeneous composition of students in terms of age and background play a role as well as the differences regarding school infrastructure. The time spans each teacher could free up for our test runs varied further. We usually tried to design a basic exercise, which given more time could be extended easily or be made more demanding. The teachers often don’t have administrative rights for the school computer workstations. To install software a request has to be sent to the system administrator a couple of weeks before the class starts. Further, the operating systems differ between schools (mostly Mac OS is used) as well as the hardware properties. Even though the EmbedITApp runs on the main three operating systems, it was easier to prepare small ASUS EeePC netbooks which we brought to each class at the testing day. Thus, we could guarantee the proper function of both hardware and software. This simple solution turned out to be a critical success factor and has also been appreciated much by the teachers.

Preparing hands-on exercises is usually very time consuming. Material costs are further an issue. It is almost impossible to cope with a large number of attendees. We found that the optimal class size is 10-12 students in groups of 2 people. The morphology exercise proved to be the perfect choice if a large number of students have to be managed. Material costs are very low, using the glue gun is self-explanatory, and the discovery learning method did not require many instructions. The preparation efforts are further very low, one has only to call and save the preprogrammed oscillation control routine of the servo module.

## 6.4 Summary

The modularity of EmbedIT enabled the design of three very different exercises with only one platform (research question 7). The control exercise trains those “traditional” engineering skills (e.g. control logic to avoid obstacles, etc.) usually covered by educational robotics initiatives. The Braitenberg vehicle exercise fosters scientific thinking. The initial thought experiment and the following experiment on the real robot helps them to verify their ideas. Further they are confronted

with the frame-of-reference problem, emergent behavior, individual level vs. collective level, etc. Other important skills are required for the morphology exercise. Students have to take material properties, dynamics, friction, etc. into consideration. There is no model or “correct” answer to their problem, they have to explore solutions in a highly creative way. In the physical computing exercise students have to deal with the properties of motors and sensors. They have to find the right application for a sensor and are challenged to come up with mechanical construction solutions. Time management and team work are further critical skills for this project. The ubiquitous computing exercise promotes the creation of novel, creative ideas in a brainstorming session. These are important skills for future innovators.

EmbedIT performed well in our real life classroom situations (research question 8). The students embraced the toolkit without any fear of contact. Even though the toolkit was in a prototyping state (with exposed PCB's), the individual modules or components rarely broke. The weakest points were the motors, which frequently broke due to jammed gears. The EmbedITApp had some usability and stability issues, which will be solved with future implementations.

Concerning the best practices (research question 9) we found that the software should work out of the box with minimal installation efforts. We solved this with the netbooks. This is of course not useful in the long run. A browser-based application could be the solution. The morphology exercise proved to be very suitable for large class sizes in terms of both preparation efforts and costs.





# Combined Survey Results

This chapter we address research question 10 of chapter 1.

### 10. *How can educational robotics classes be evaluated?*

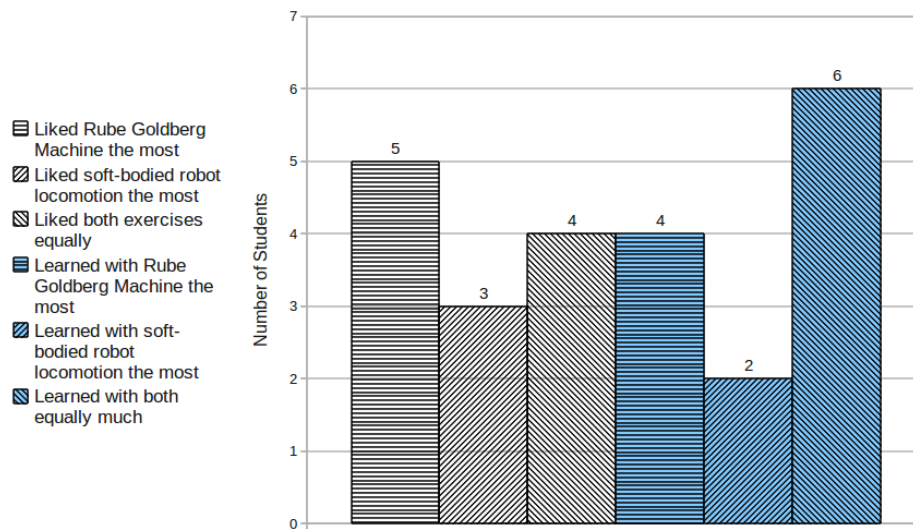
We used questionnaires to collect feedback from the students about their background, the class itself, and the toolkit. Due to the low number of students in each class, we mainly used the collected data to get a qualitative idea, which always has to be looked at in combination with our observations and personal conversations. This section summarizes the survey results of the combined data of each individual class.

It would be interesting to know which of the three exercises is the most popular one among the students. Ideally, every class would do all types of exercises and then compare them with one another. Unfortunately, this was not possible in the scope of this thesis. We could only conduct two different exercises (Rube Goldberg Machine and soft-bodied robot locomotion) at the same time with the Kantonsschule Trogen. We asked the twelve students to vote which exercise they liked the most and in which they think they had learned the most. The results are shown in Figure 7.1. The Rube Goldberg Machine was liked the most in this class, and they had learned equally much with both exercises. We are pleased by this outcome, however, the number of students is too low to draw a conclusion based on this voting.

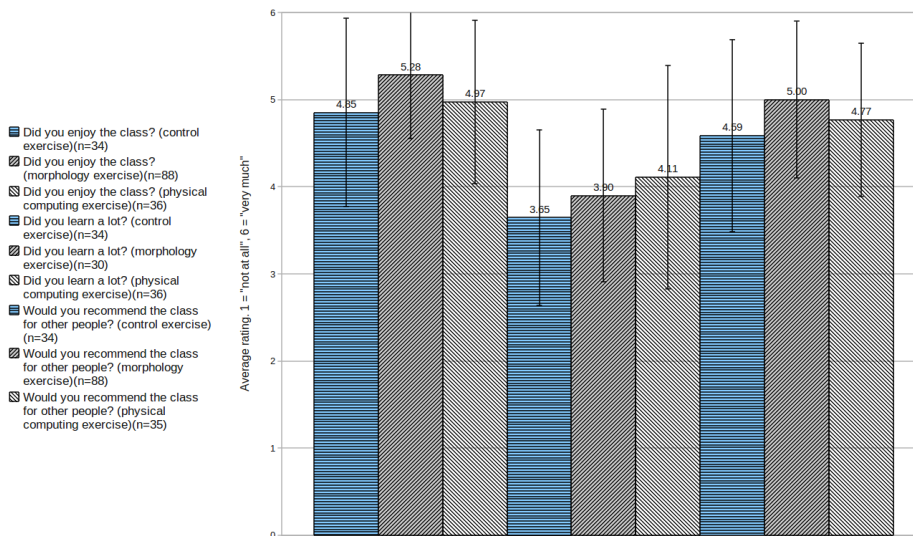
In the questionnaires of all exercise types we asked similar questions such as “did you enjoy the class?”, “have you learned a lot?”, or “would you recommend the class for other people?”. Further, we asked where applicable whether they found the EmbedIT software user-friendly or if they preferred to work with the LEGO Mindstorms platform. Figure 7.2 and Figure 7.3 combine the survey results from Figures 6.6, 6.11, 6.13, 6.17, 6.19, 6.20, and the data from [Assaf et al., 2012b].

The number of students rated for each question differ in Figure 7.2 due to the inhomogeneous composition of each class and the fact that we have not conducted each exercise equally often. Nevertheless, the plot can give us some interesting qualitative information about the exercises. All three exercises were basically equally enjoyed, the same constellation is reflected in the recommendation question. Also the learning factor seems equally distributed among all three exercises, however it is usually lower than the “fun” factor.

Figure 7.3 shows the combined survey results concerning the EmbedIT toolkit. It can be seen that the graphical programming was overall appreciated, though the user-friendliness of the GUI has still room for improvement. LEGO Mindstorms and EmbedIT share a similar rating as a preferred toolkit, however, the error bar for LEGO Mindstorms is much larger, therefore preferences vary significantly among students.



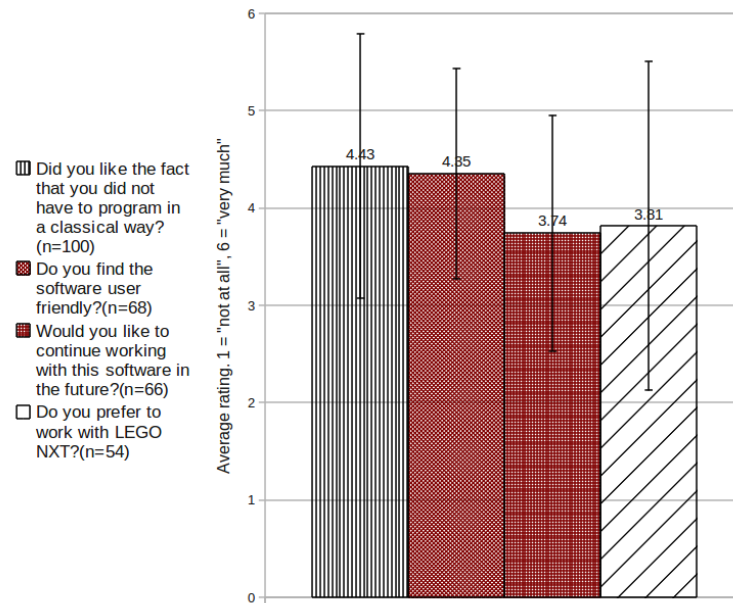
**Figure 7.1:** Twelve students from Kantonsschule Trogen compared the two exercises Rube Goldberg Machine and soft-bodied locomotion. We asked which exercise they liked the most and in which they think they learned the most. The plots show the number of students which voted for either Rube Goldberg Machine, soft-bodied locomotion exercise, or both equally.



**Figure 7.2:** Combined survey results of the data concerning the exercises from Figures 6.6, 6.11, 6.13, 6.17, 6.19, 6.20, and the data from [Assaf et al., 2012b]. The scale ranges between one and six, where one stands for "not at all" and six represents "very much".

## 7.1 Summary

The survey results give us an impression how students liked the exercises, however, no conclusions can be drawn based on this data. There are no statistically significant differences between



**Figure 7.3:** Combined survey results of the data concerning the EmbedIT toolkit from Figures 6.6, 6.11, 6.13, 6.17, 6.19, 6.20. The scale ranges between one and six, where one stands for "not at all" and six represents "very much".

outcomes of the individual questions. We always looked at this data together with qualitative feedback we received while teaching the classes. Our experience was that we acquired much more information by simply observing students' behaviors as well as through personal conversations with students and teachers. As mentioned previously in section 2.5.3 of this thesis, the evaluation of classes (research question 10) is an open issue in educational robotics research. It is crucial to work more closely together with pedagogues in the future. They are the experts to choose the right method to collect data. They have further the experiences to compose suitable questionnaires and to interpret the data.



# Discussion and Conclusion

In this thesis, we aimed to contribute to science and technology education by developing a novel open-source robotic kit (EmbedIT). The toolkit should enable students to explore in an motivating and fun way interdisciplinary topics such as (biologically inspired) robotics, embodied artificial intelligence, electronics, computer science, neuroscience, psychology, arts, etc. We designed a number of example exercises using this toolkit which train a variety of skills young people need for a successful professional life. Educational robotics is an interdisciplinary research field. It requires pedagogues, engineers, and teachers to collaborate closely together (Figure 1.1). The insights gained by addressing ten research questions (chapter 1) in the context of this thesis contribute to all of these three expert groups. There are conceptual questions, technical issues, questions about teachers and classroom practices as well as questions addressing evaluation issues.

On the conceptual level (research question 1–2) our modular design approach with re-usable, generic PCB's proved to be right for the requirements of versatility, adaptability, and scalability. So were our design choices (e.g. distributed control, preprogrammed control routines, standard interfaces to attach unconventional materials) to enable exploration of biologically inspired robotics and embodied artificial intelligence. We had to solve a number of technical challenges (research question 3–4). We achieved modularity by developing three different modules (master, actuator, sensor), each of which possess a generic design. The CAN bus architecture enabled the required plug-and-play functionality. The main questions raised by designing this modular toolkit were level of modularity (e.g. a group of the same sensors or actuators are defined a module), communication protocol, and functionality of the graphical user interface (e.g. user can tune preprogrammed oscillators). The compliance with industrial standards together with the policy to incorporate solely open-source projects enabled easy customization of the platform. LEGO Mindstorms is the most prevalent robotic platform in Swiss schools (this also applies to Europe and USA). We see EmbedIT as a complementary kit to LEGO Mindstorms and not as a competitive one. However, to identify the quality and the potential of our toolkit, it makes sense to compare it with the state-of-the-art product. In Table 8.1 we are comparing the for us important features with EmbedIT and LEGO Mindstorms. Based on the comparison we see that the main EmbedIT drawbacks at the moment are the software stability, hardware stability, missing on-line community, and limited graphical programming functionality. The stability of hardware and software is valued as an important factor for the teachers, so this is surely a high priority open issue. Nevertheless, these are all issues that will improve with a continuing development and are not based on a flawed concept.

Our aim was to get insights about classroom practices of secondary school teachers that are giving computer science and engineering classes (research question 5–6). Based on the teacher interviews we learned that each teacher prefers to develop his own class syllabus. The topics covered in the EFI classes differ a lot from teacher to teacher. So are the preferences for educational

toolkits, some teachers are not willing to deal with low-level electronics opposed to others who see it as an educational gain. They argue that even if a student shortens a circuit it is a valuable insight. Especially nowadays, where the electronics of all technological devices are covered by neatly designed casings to ensure flawless operation. The fact, that the electronics of the EmbedIT toolkit is not hidden in a casing and therefore is not functioning as a “black box”, was mentioned frequently as very positive. However, this contradicts the clear preference of the teachers to have stable hardware to work with. The EmbedIT PCB’s will have to be protected in some kind of casing, maybe a transparent one would meet both requirements.

We designed three different hands-on exercises (focus on control, morphology, physical computing) in order to test the versatility of the toolkit (research question 7). The aim was to train a variety of skills with just one platform. The exercises were conducted in real class environments with total 179 students. These test runs brought valuable insights on the usability and stability of EmbedIT (research question 8), enabled the definition of best practices (research question 9), and the collection of evaluation data (research question 10). The morphology exercise was a big success in respect to survey results, costs and preparation efforts. Thanks to the preprogrammed motor oscillator function provided by the toolkit, sine parameters could be explored and saved easily on the robot in real time. The lightweight actuator board enabled the construction of soft-bodied robots without influencing the locomotion dynamics significantly. The outcome of these classes resulted in a much greater variety of diverse robotic creatures and locomotion patterns than previously seen in the same exercise concept with the use of LEGO Mindstorms. The third exercise, the physical computing scenario received also very positive feedback. The students explicitly mentioned in the free comment section of the questionnaire the appreciation of the creative work by building the parts of the Rube Goldberg machine. This was the most open and unconstrained exercise which was especially demanding for the EmbedIT hardware since a large number of different sensors and actuators had to be integrated within only a few modules. The toolkit was proven very suitable for this kind of open exercise.

We observed generally a great engagement of the students in all three exercises, especially in those which involved constructing a robot or an interactive system. Interestingly, the learning factor was always lower than the enjoyment factor. This might be explained by the single instance of each class given and its short duration. We simply did not have the time to provide profound background information on the technology as well as on theoretical concepts. The morphology exercise, for instance, clearly would require an introduction to biologically inspired robotics, biomechanics, or embodied artificial intelligence. This way the exercise concept might have been introduced out of context with respect to the students’ current curriculum. Therefore, they did not really understand the global vision behind this exercise. It has to be mentioned, however, that the morphology exercise was very popular among the teachers. Some explicitly asked me to conduct that one during my visit. Another teacher is planning to integrate it in his own curriculum in the future.

We contribute with this work to the interdisciplinary research field of educational robotics and to the dissemination of biologically inspired robotics and embodied artificial intelligence. Each expert group (pedagogues, engineers, teachers) benefits further from the insights gained, the tools developed, and the example exercises designed. The exercises could either be directly adopted by the teachers or customized accordingly. They can further use EmbedIT to design other exercises. The best practices are additionally beneficial for the teachers. Our design concepts, the information about the teachers as well as the implementation of the platform contribute to the engineers. Due to the open-source license they can extend and customize the platform according to their needs. Our technical design approach can give important recommendations for any toolkit developer. Researchers such as pedagogues, social scientists, engineers or psychologists can use this platform as an user-friendly tool to conduct further qualitative and quantitative studies in the context of educational (robotics) research.

**Table 8.1:** Feature comparison between LEGO Mindstorms and EmbedIT

Feature	LEGO Mindstorms	EmbedIT
Flexibility of robot bodies that can be built	✓, by using LEGO blocks or Tetrix	✓, by using any mechanical construction kit
Quick and easy assembly	✓, plug and play functionality	✓, plug and play functionality
Solderless connections	✓, by using plugs	✓, by using plugs
Stability of software	✓	✗, not yet but improving
Stability of hardware	✓	✗, not yet, PCB's have to be protected with a casing
Large on-line community	✓	✗, Not yet
Graphical programming	✓, LEGO software or Lab-View	✓, yet limited
Textual programming	✓, Java, C	✓, C/C++
A variety of sensors	✓, touch, light, sound, ultrasonic distance, compass, color, RF ID, and some more expensive ones manufactured by HiTechnic <sup>1</sup>	✓, supports all LEGO sensors and even more sensors for a much cheaper price (because they can be purchased in any shop)
Number of sensors attachable	4	8 for each sensor board and several sensor boards within a system
A variety of motors	✗, only LEGO full rotary servo motor	✓, supports LEGO motor and even more actuators (e.g. vibration motor, DC motor, standard servo motor)
Number of actuators attachable	3	4 for each actuator board and a number of actuator boards within a system
Communication to PC	✓, Bluetooth, USB	✓, Bluetooth, USB, RS232
Good quality of sensors and actuators	✓, however, user is restricted to what LEGO provides	✓, any sensor and actuator can be attached, therefore it's up to the user
Mechanical stability of construction parts (limited clearance etc.)	✗, unless Tetrix is used	✓, any mechanical construction kit can be used.
Visibility of the electronics	✗, hidden in casing	✓
Open-source license	✗	✓
Industrial standards	✗, e.g. proprietary hole-spacing, hole-size, plugs, I <sup>2</sup> C interface	✓, metric hole-size, commonly used plugs, and communication technologies (e.g. CAN, RS232)
Graphical access of sensor data	✗	✓
Graphical remote control functionality	✗	✓
Preprogrammed routines (e.g. CPG)	✗	✓
Compatibility with other popular toolkits	✗	✓
Possibility to work with soft-bodied robots	✗, central controller is too heavy	✓, lightweight modules
Distributed control, modularity	✗	✓





# Future Work

In the previous chapters we saw the great potential of the EmbedIT toolkit. A first stable release should be scheduled, with the possibility of users to purchase hardware components, starting with PCB's. Hardware schematics and software source will be released as open-source. A number of complete toolkits consisting of EmbedIT modules, a variety of sensors and actuators, as well as mechanical construction parts should be distributed as a trial kit to interested teachers. The teacher will need an introduction on how to use the kit together with our example exercises. The teachers will most likely adapt the exercises to their needs or even create completely new ones. These trial users will provide more evaluation data about the toolkit and further generate more case studies on its applicability. To get to this stage, future work aims to:

- Improve the GUI's user-friendliness.
- Improve the GUI such that it runs more stably.
- Extend the graphical programming functionality.
- Create transparent casings for EmbedIT modules to prevent short-circuits.
- Document and release software and hardware to facilitate user contributions. Choose a suitable open-source license.
- Produce a number of toolkits to be distributed to further trial participants.
- Build a community website with EmbedIT manuals, information on toolkits, links, videos, tutorials, etc.
- Maintain and increase the network of teachers and educational robotics researchers. Further encourage them to share their ideas and experiences to become part of a contributing community.

To improve the GUI it would be valuable to work closely with interface and/or interaction designers. Conversations with interface experts have already generated ideas for improvements. We list possible extensions and new features for a long-term perspective of this project which would make the toolkit a powerful platform that goes beyond teaching, for example as rapid prototyping platform for research and industry.

- Integration of the Arduino programming interface ("processing", based on Java) to the EmbedITApp to enable easy switching from graphical to textual programming.

- Possible integration of the graphical “scratch” programming interface to the EmbedITApp to enable an intermediate step to pure textual programming.
- Integration of the wireless XBee technology as an alternative to the wired CAN bus to connect the modules. This would result in a truly modular sensor and actuator network system.
- Implementation of support to run the EmbedITApp on smart phones and tablets.
- Support to send notifications, e.g. sensor data, by email or to social media platforms such as Facebook, Twitter, etc. (as done in the Botanicalls<sup>1</sup> project).
- Integration of an open-source simulation software for virtual experimentation.
- Increasing the repertoire of preprogrammed control routines (e.g. more CPG’s, learning algorithms, etc.).
- Integration of video (e.g. attaching a Gumstix<sup>2</sup> with a camera to the bus including tracking routines, etc., interfacing with the motion sensing device Kinect<sup>3</sup>).
- Automatic export of collected sensor data for graphical visualization (e.g. with GNU Octave).

In this thesis we created design concepts, solved technical challenges, evaluated classes, gained insights into teaching practices of secondary school teachers, and defined best practices for hands-on exercises in classrooms. We were able to show that the toolkit is suitable for a variety of teaching purposes and that it is easily competitive with state-of-the-art products such as LEGO Mindstorms, which further holds extensive potential. EmbedIT is a starting point for further development and hopefully, by means of an contributing community it has the potential to become the state-of-the-art product which modern teaching environments require.

---

<sup>1</sup><http://www.botanicalls.com/>

<sup>2</sup><http://www.gumstix.com/>

<sup>3</sup><http://www.xbox.com/en-US/KINECT>

---

# Bibliography

- [Alimisis, 2012] Alimisis, D. (2012). Robotics in education & education in robotics: Shifting focus from technology to pedagogy. In *Proceedings of the 3rd International Conference on Robotics in Education (RiE 2012)*, pages 7–14.
- [Alimisis et al., 2010] Alimisis, D., Arlegui, J., Fava, N., Frangou, S., Ionita, S., Menegatti, E., Monfalcon, S., Moro, M., Papanikolaou, K., and Pina, A. (2010). Introducing robotics to teachers and schools: experiences from the terecop project. In *Proceedings of the Constructionism 2010 Conference*, Paris, France.
- [Alimisis et al., 2007] Alimisis, D., Moro, M., Arlegui, J., Pina, A., Frangou, S., and Papanikolaou, K. (2007). Robotics & constructivism in education: the terecop project. In *Proceedings of the 11th European Logo Conference*, pages 1–11. Comenius University Bratislava.
- [Apiola et al., 2010] Apiola, M., Lattu, M., and Pasanen, T. A. (2010). Creativity and intrinsic motivation in computer science education: experimenting with robots. In *Proceedings of the fifteenth annual conference on Innovation and technology in computer science education, ITiCSE '10*, pages 199–203, New York, NY, USA. ACM.
- [Assaf et al., 2012a] Assaf, D., Larsen, J. C., and Reichardt, M. (2012a). Extending mechanical construction kits to incorporate passive and compliant elements for educational robotics. In *Proceedings of 3rd International Conference on Robotics in Education (RiE 2012)*, pages 33–40.
- [Assaf and Pfeifer, 2011a] Assaf, D. and Pfeifer, R. (2011a). Embedit – an open robotic kit for education. In *Research and Education in Robotics – EUROBOT 2011, Communications in Computer and Information Science*, volume 161, pages 29–39. Springer.
- [Assaf and Pfeifer, 2011b] Assaf, D. and Pfeifer, R. (2011b). Robotics as part of an informatics degree program for teachers. In *Proceedings of Society for Information Technology & Teacher Education International Conference 2011*, pages 3128–3133. Chesapeake, VA: AACE.
- [Assaf et al., 2012b] Assaf, D., Yu, X., Wang, L., and Iida, F. (2012b). Teaching soft-robotics: A case study in building elastic creatures that locomote. *submitted*.
- [Balch et al., 2008] Balch, T., Summet, J., Blank, D., Kumar, D., Guzdial, M., Walker, D., Sweat, M., Gupta, G., Tansley, S., Eilbert, N., and Gavin, A. (2008). Designing personal robots for education: Hardware, software, and curriculum. *Pervasive Computing, IEEE*.
- [Barbero et al., 2011] Barbero, A., Demo, B., and Vaschetto, F. (2011). A contribution to the discussion on informatics and robotics in secondary schools. In *INNOC - Austrian Society for Innovative Computer Sciences*, pages 201–206.

- [Bennett et al., 2008] Bennett, S., Maton, K., and Kervin, L. (2008). The digital natives" debate: A critical review of the evidence. *British Journal of Educational Technology*, Vol. 39 No 5.
- [Bers, 2008] Bers, M. U. (2008). *Blocks to Robots: Learning with Technology in the Early Childhood Classroom*. NY: Teachers College Press.
- [Blank, 2006] Blank, D. (2006). Robots make computer science personal. *Communications of the ACM*, 49(12):25–27.
- [Blank and Kumar, 2010] Blank, D. and Kumar, D. (2010). Assessing the impact of using robots in education, or: How we learned to stop worrying and love the chaos. In *Educational Robotics and Beyond: Design and Evaluation*, AAAI Spring Symposium Series, pages 3–7. AAAI Press.
- [Braitenberg, 1986] Braitenberg, V. (1986). *Vehicles, experiments in synthetic psychology*. MIT Press, Cambridge Massachusetts.
- [Bredenfeld et al., 2010] Bredenfeld, A., Hofmann, A., and Steinbauer, G. (2010). Robotics in education initiatives in europe – status, shortcomings and open questions. In *Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPARG 2010)*, pages 568–574.
- [Bredenfeld and Leimbach, 2010] Bredenfeld, A. and Leimbach, T. (2010). The roberta initiative. In *Workshop Proceedings of SIMPAR 2010 Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots*, pages 558–567.
- [Church et al., 2010] Church, W., Ford, T., Perova, N., and Rogers, C. (2010). Physics with robotics: Using lego mindstorms in high school education. In *Educational Robotics and Beyond: Design and Evaluation*, AAAI Spring Symposium Series, pages 47–49. AAAI Press.
- [DeMichele et al., 2008] DeMichele, M. S., Demo, G. B., and Siega, S. (2008). A piedmont school-net for a k-12 mini-robots programming project: Experiences in primary schools. In *Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2008)*, pages 90–99.
- [Fagin and Merkle, 2002] Fagin, B. S. and Merkle, L. (2002). Quantitative analysis of the effects of robots on introductory computer science education. *Journal of Educational Resources in Computing*, 2(4).
- [Faludi, 2010] Faludi, R. (2010). *Building Wireless Sensor Networks: with ZigBee, XBee, Arduino, and Processing*. O'Reilly Media.
- [Frاند, 2000] Frاند, J. (2000). The information-age mindset: changes in students and implications for higher education. *EDUCAUSE*, 35:14–24.
- [Frangou et al., 2010] Frangou, S., Papanikolaou, K., Aravecchia, L., Montel, L., Ionita, S., Arlegui, J., Pina, A., Menegatti, E., Moro, M., Fava, N., Monfalcon, S., and Pagello, I. (2010). Representative examples of implementing educational robotics in school based on the constructivist approach. In *Workshop Proceedings of SIMPAR 2010 Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots*, pages 54–65.
- [Hlinovsky and Polcar, 2010] Hlinovsky, M. and Polcar, T. (2010). Subject robots at the ctu fee in prague – using lego robots to teach the fundamentals of feedback control. In *Proceedings of 1st International Conference on Robotics in Education (RiE 2010)*, pages 71–76. FEI STU, Slovakia.
- [Hoffmann and Pfeifer, 2012] Hoffmann, M. and Pfeifer, R. (2012). The implications of embodiment for behavior and cognition: animal and robotic case studies. *CoRR*, abs/1202.0440.

- [Hofmann and Steinbauer, 2010] Hofmann, A. and Steinbauer, G. (2010). The regional center concept for robocupjunior in austria. In *Proceedings of 1st International Conference on Robotics in Education (RiE 2010)*, pages 77–82. FEI STU, Slovakia.
- [Howe and Strauss, 2000] Howe, N. and Strauss, W. (2000). *Millenials rising: the next great generation*. New York: Vintage.
- [Howe and Strauss, 2003] Howe, N. and Strauss, W. (2003). *Millenials go to college*. Washington DC: American Association of Collegiate Registrars and Admission Officers.
- [Hung et al., 2012] Hung, V. Q., Jafari, A., Iida, F., and Pfeifer, R. (2012). The effects of stiffness adjustability to cost of transport of single-leg hopping. *submitted*.
- [Igoe, 2007] Igoe, T. (2007). *Making Things Talk: Practical Methods for Connecting Physical Objects*. Make.
- [Iida et al., 2005] Iida, F., Gomez, G., and Pfeifer, R. (2005). Exploiting body dynamics for controlling a running quadruped robot. In *Proceedings of the 12th International Conference on Advanced Robotics, ICAR (2005)*.
- [Imberman and Klibaner, 2005] Imberman, S. P. and Klibaner, R. (2005). A robotics lab for cs1. *Journal of Computing Sciences in Colleges*, 21(2):131–137.
- [Kabátová and Pekárová, 2010] Kabátová, M. and Pekárová, J. (2010). Lessons learnt with lego mindstorms: from beginner to teaching robotics. *AT&P journal PLUS2*, pages 51–56.
- [Kandlhofer et al., 2012] Kandlhofer, M., Steinbauer, G., Sundström, P., and Weiss, A. (2012). Evaluating the long-term impact of robocupjunior: A first investigation. In *Proceedings of the 3rd International Conference on Robotics in Education (RiE 2012)*, pages 87–94.
- [Kay, 2010] Kay, J. (2010). Robots as recruitment tools in computer science: The new frontier or simply bait and switch? In *Educational Robotics and Beyond: Design and Evaluation*, AAAI Spring Symposium Series, pages 26–30. AAAI Press.
- [Kennedy et al., 2006] Kennedy, G., Krause, K., Judd, T., Churchward, A., and Gray, K. (2006). First year students’ experiences with technology: are they really digital natives? *Australian Journal of Educational Technology*.
- [Kvavik, 2005] Kvavik, R. (2005). Convenience, communication, and control: How students use technology. *Educating the net generation*, pages 7.1–7.20.
- [Markham and King, 2010] Markham, S. A. and King, K. N. (2010). Using personal robots in cs1: experiences, outcomes, and attitudinal influences. In *Proceedings of the 15th annual conference on Innovation and technology in computer science education, ITiCSE ’10*, pages 204–208, New York, NY, USA. ACM.
- [McEuen, 2001] McEuen, S. (2001). How fluent with information technology are our students? *EDUCAUSE*, Vol. 4.
- [Melchior et al., 2005a] Melchior, A., Cohen, F., Cutter, T., and Leavitt, T. (2005a). More than robots: An evaluation of the first robotics competition participant and institutional impacts. Technical report, Heller School for Social Policy and Management, Brandeis University, Waltham, MA.
- [Melchior et al., 2005b] Melchior, A., Cutter, T., and Cohen, F. (2005b). Evaluation of first lego league underserved initiative. Technical report, Heller School for Social Policy and Management, Brandeis University, Waltham, MA.

- [Nourbakhsh et al., 2005] Nourbakhsh, I., Crowley, K., Bhavé, A., Hamner, E., Hsiu, T., Perez-Bergquist, A., Richards, S., and Wilkinson, K. (2005). The robotic autonomy mobile robotics course: Robot design, curriculum design and educational assessment. *Autonomous Robots*, 18(1):103–127.
- [Oliver and Goerke, 2007] Oliver, B. and Goerke, V. (2007). Australian undergraduates' use and ownership of emerging technologies: implications and opportunities for creating engaging learning experiences for the net generation. *Australian Journal of Educational Technology*, 23, 2:171–186.
- [O'Sullivan and Igoe, 2004] O'Sullivan, D. and Igoe, T. (2004). *Physical Computing: Sensing and Controlling the Physical World with Computers*. Thomson.
- [Patterson, 2005] Patterson, D. A. (2005). Restoring the popularity of computer science. *Communications of the ACM*, 48(9):25–28.
- [Paul et al., 2000] Paul, C., Hafner, V., and Bongard, J. (2000). Teaching new artificial intelligence using constructionist edutainment robots. Technical report, Artificial Intelligence Laboratory, Department of Informatics, University of Zurich.
- [Pekárová, 2008] Pekárová, J. (2008). Using a programmable toy at preschool age: Why and how? In *Workshop Proceedings of SIMPAR 2008 Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots*, pages 112–121.
- [Petre and Price, 2004] Petre, M. and Price, B. (2004). Using robotics to motivate 'back door' learning. *Education and Information Technologies*, 9(2):147–158.
- [Petrovic et al., 2010] Petrovic, P., Balogh, R., and Lucny, A. (2010). Robotika.sk approach to educational robotics from elementary schools to universities. In *Proceedings of 1st International Conference on Robotics in Education (RiE 2010)*, pages 165–170. FEI STU, Slovakia.
- [Pfeifer and Bongard, 2006] Pfeifer, R. and Bongard, J. (2006). *How the Body Shapes the Way We Think – A New View of Intelligence*. MIT Press, Cambridge Massachusetts.
- [Pfeifer et al., 2007] Pfeifer, R., Lungarella, M., and Iida, F. (2007). Self-organization, embodiment, and biologically inspired robotics. *Science*, 318:1088–1093.
- [Pfeifer and Scheier, 1999] Pfeifer, R. and Scheier, C. (1999). *Understanding Intelligence*. MIT Press.
- [Piaget, 1952] Piaget, J. (1952). *The origins of intelligence in children*. New York: International Universities Press.
- [Piaget, 1954] Piaget, J. (1954). *The construction of reality in the child*. New York: Basic Books.
- [Pils, 2010] Pils, W. (2010). Skaty – a skating robot. Master's thesis, Artificial Intelligence Laboratory, Department of Informatics, University of Zurich.
- [Pittí et al., 2010] Pittí, K., Curto, B., García, J., and Moreno, V. (2010). Nxt workshops: Constructionist learning experiences in rural areas. In *Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010)*, pages 504–513.
- [Prensky, 2001a] Prensky, M. (2001a). Digital natives, digital immigrants. *On the Horizon*, 9,5:1–6.
- [Prensky, 2001b] Prensky, M. (2001b). Digital natives, digital immigrants, part ii. do they really think differently? *On the Horizon*, 9,6:1–6.

- [Rocard et al., 2007] Rocard, M., Csermely, P., Jorde, D., Lenzen, D., Walberg-Henriksson, H., and Hemmo, V. (2007). Science education now: a renewed pedagogy for the future of europe. *RTD info*, EUR22845.
- [Sagan, 1990] Sagan, C. (1990). Why we need to understand science. *Skeptical Inquirer*, 14.3.
- [Saranli et al., 2001] Saranli, U., Buehler, M., and Koditschek, D. E. (2001). Rhex: A simple and highly mobile hexapod robot. *Journal of Robotics Research*, 20, 7:616631.
- [Savery and Duffy, 1996] Savery, J. and Duffy, T. (1996). Problem based learning: An instructional model and its constructivist framework. *Educational Technology*.
- [Simon, 1976] Simon, H. (1976). *Administrative behavior: A study of decision-making processes in administrative organization*. New York: Free Press, 3rd edition.
- [Sklar et al., 2003] Sklar, E., Eguchi, A., and Johnson, J. (2003). *RoboCupJunior: Learning with Educational Robotics*, volume 2752 of *Lecture Notes in Computer Science*, pages 238–253. Springer Berlin Heidelberg.
- [Sterchi, 2011] Sterchi, D. (2011). User interface – analyse, design und evaluation von graphical user interfaces. Master’s thesis, Artificial Intelligence Laboratory, Department of Informatics, University of Zurich.
- [Stoeckelmayr et al., 2011] Stoeckelmayr, K., Tesar, M., and Hofmann, A. (2011). Kindergarten children programming robots: A first attempt. In *Proceedings of 2nd International Conference on Robotics in Education (RiE 2011)*, pages 185–192. INNOC – Austrian Society for Innovative Computer Sciences.
- [Tapscott, 1998] Tapscott, D. (1998). *Growing up digital: the rise of the Net generation*. New York: McGraw-Hill.
- [Visser and Burkhard, 2007] Visser, U. and Burkhard, H. (2007). Robocup: 10 years of achievements and future challenges. *AAAI AI Magazine*, 28, 2.
- [Wyffels et al., 2011] Wyffels, F., Bruneel, K., Kindermans, P., D’Haene, M., Woestyn, P., Bertels, P., and Schrauwen, B. (2011). Robot competitions trick students into learning. In *Proceedings of 2nd International Conference on Robotics in Education (RiE 2011)*, pages 47–52. INNOC – Austrian Society for Innovative Computer Sciences.
- [Wyffels et al., 2010] Wyffels, F., Hermans, M., and Schrauwen, B. (2010). Building robots as a tool to motivate students into an engineering education. In *Proceedings of 1st International Conference on Robotics in Education (RiE 2010)*, pages 49–52. FEI STU, Slovakia.





## Appendix A

---

# Robotics as Part of an Informatics Degree Program for Teachers

Reprinted from:

Assaf, D. and Pfeifer, R. (2011). *Robotics as Part of an Informatics Degree Program for Teachers*, In Proceedings of Society for Information Technology & Teacher Education International Conference 2011, pages 3128–3133. Chesapeake, VA: AACE.

## Robotics as Part of an Informatics Degree Program for Teachers

Dorit Assaf  
Artificial Intelligence Laboratory  
Department of Informatics  
University of Zurich  
Switzerland  
assaf@ifi.uzh.ch

Rolf Pfeifer  
Artificial Intelligence Laboratory  
Department of Informatics  
University of Zurich  
Switzerland  
pfeifer@ifi.uzh.ch

**Abstract:** This paper describes two robot competitions that took place within a robotics class for teachers. The robotics class was part of a two-year master's degree program that aims at educating upper secondary school teachers of different backgrounds in informatics, a discipline that is not yet a mandatory part of the school curriculum in Switzerland. The aim of these robot competitions was to familiarize the teachers with robotic hardware and software such that they would be able to design their own informatics class syllabus. We describe the robotic platforms used, the competitions, their aims and results. Furthermore, we address the question whether robots are a suitable tool for teacher education in informatics.

### Introduction

There has been a rapid development of information and communication technology in the last few decades that highly influenced our society. Informatics is seen as the science that drives the progress in these technologies. In order to maintain economic competitiveness, education in this discipline is crucial. Informatics is not yet a part of the curriculum of upper secondary schools in Switzerland. The Swiss school authority has recently identified this and decided to incorporate informatics as an optional discipline in its school curriculum. Consequently, teacher education in informatics is an urgent requirement. The Swiss Hasler foundation has funded a degree program 'Master of Advanced Studies Informatics in Upper Secondary Schools' to address this need. This program targets teachers that are willing to teach informatics in addition to their primary discipline. This two-year master's study program conducted by several Swiss Universities includes topics such as programming and algorithms, Internet and multimedia, simulation, databases and information systems, theoretical informatics, and robotics. Since there is no official syllabus for this optional discipline in informatics in upper secondary schools provided by the school authority, this broad background of informatics should enable the teachers to produce the syllabus by themselves. Within these two years the teachers attend classes on Fridays and Saturdays. In addition to the study program costs the foundation partly covered the costs for the teacher's substitutes at school during their absence. The teachers had to apply to be accepted for this study program. The program committee interviewed each applicant individually and decided based on criteria such as background and motivation. Teachers with different backgrounds were targeted in order to avoid a too strong affiliation of informatics with only one discipline such as Mathematics. The aim is to make informatics an appealing alternative choice for those students that neither want to focus on humanities nor on natural sciences.

This paper describes two exemplary robot exercises that were conducted in the 'Robotics' course taught by the authors, as part of this teacher education program in January 2010 and 2011. The teachers were between the age of 32 and 54 and familiar with very different disciplines. Fifteen male teachers attended the first class in 2010 (4 Mathematics, 3 Informatics, 2 Physics, 1 Chemistry, 1 German literature, 1 Biology, 1 Sports, 1 Greek, 1 Economics). Twenty teachers attended the second round in 2011 among them 4 females (5 Mathematics, 5 Physics, 3 Biology, 3 Sports, 1 Chemistry, 1 Economics, 1 Geography, 1 Arts). The informatics teachers used to teach only

applications such as Microsoft Office and how to use the Internet. With that study program they wanted to broaden their knowledge. In this class, robots were used as a learning tool for the teachers. As robots are often attractive tools for hands-on education of younger students in science and mathematics due to the 'fun' factor, we believe that it is necessary for the teachers to learn and have first-hand experiences with robots themselves. In addition, since the teachers will have to design their future informatics class by themselves the aim was to familiarize them with the 'inside' of a robot. They should have some basic knowledge of the components which are necessary to build a robot and how they function. Therefore, we built and provided a custom robotic platform for one of the robot competitions. This knowledge should enable them to evaluate off-the-shelf robot kits in order to choose the right one for their own purposes.

In the rest of this paper, besides the related work, two robot exercises, the LEGO NXT and predator & prey robot competitions, are described. This is followed by results, discussion and conclusion.

## **Related Work**

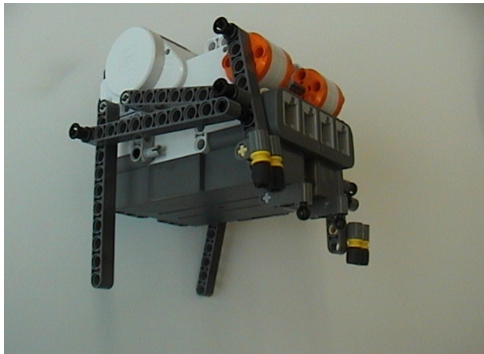
The number of students showing interest in pursuing a career in Science, Technology, Engineering and/or Math (STEM) is decreasing in the USA (Miller et al. 2010) as well as in Europe (Rocard et al. 2007). This phenomenon has gained a lot of attention from teachers, researchers, politicians, authorities and other stakeholders. Consequently, a vast number of reform curricula and initiatives have been developed to improve the situation, but definitive solutions have not yet emerged. Robots have been used in the last decade to introduce kids and especially girls (Bredenfeld et al. 2010) to science and technology (Alimisis et al. 2007). Class activities with robots range from kindergarten to upper secondary school. A popular educational approach to motivate young people in learning is based on the constructivist / constructionist paradigm, where learning through play can contribute to the construction of knowledge (Piaget 1952, 1954). Active learning environments through the use of interactive lessons, friendly competitions, and trial and error are therefore preferred (Sardone 2010). A competition provides additional extrinsic motivation for the students, it increases the group work skills and encourages them to identify and evaluate a variety of opinions. A large number of robot competitions emerged such as the FIRST Lego League or RoboCupJunior, all with the aim to engage young people in these disciplines (Bredenfeld et al. 2010). A widely used robotic platform for educational robotics is the LEGO NXT (Kabátová et al. 2010, Menegatti et al. 2010). Other educational robotic platforms (Balogh 2010) use the popular Arduino boards (Arduino).

## **Two Robot Competitions**

The robotics class duration was in total eight full days excluding homework hours (4 ECTS). The class syllabus covered research oriented topics in Artificial Intelligence (AI) such as 'Introduction to AI', 'Morphological Computation', 'Neural Networks', 'Artificial Evolution', 'Artificial Life' as well as topics in robotics (robot hardware, sensors and actuators). Next to many practical exercises such as the LEGO NXT robot competition the main group project that was graded was the 'predator & prey' robot competition. Even though not all of the topics covered in this robotics class were useful for the teacher's project work, the insight of state-of-the-art research in AI and robotics was appreciated (the above mentioned topics were not examined and therefore not part of the grade).

### **The LEGO NXT Robot Competition**

The goal of this robot competition was to show the influence of a robot's morphology on its performance (Pfeifer et al. 2006). A legged LEGO NXT robot with a simple control program (two motors turn forwards in constant speed) had been provided to the teachers (Figure 1a). With this setup the robot is able to walk forwards slowly. The teacher's task was to modify the robot's body such that it achieves a faster locomotion while the control program stays unchanged. This exercise setup is inspired by Rinderknecht et al. (2007). Figure 1b shows the teachers with their individual robot morphologies at the starting line of the competition.



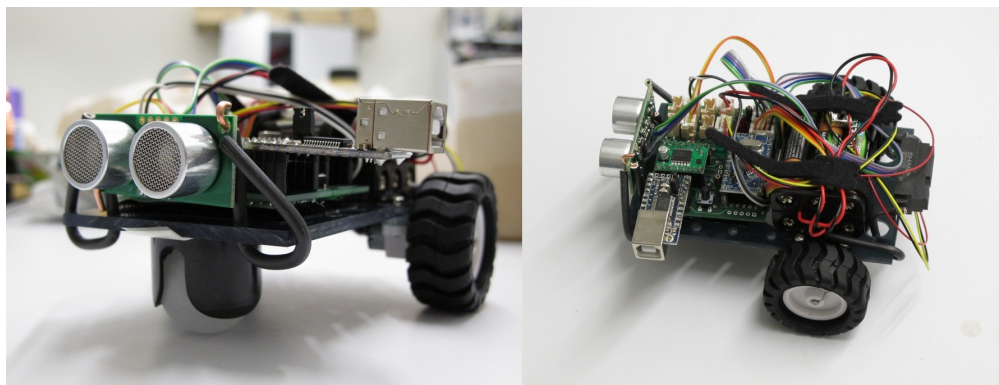
**Figure 1a:** The default LEGO NXT robot morphology provided to the teachers. Both motors turn forwards in constant speed. With this setup the robot is able to walk forwards slowly.



**Figure 1b:** The teachers with their individual robot morphologies at the starting line of the competition.

### The Predator & Prey Robot Competition

A self-made robotic platform was used for that purpose. Each group consisting of two people received two identical robots – a predator and a prey (Figure 2). The robot has two wheels actuated by two DC motors. Additionally, it was equipped with odometry, ultra sonic, light and touch sensors. In order to give the prey a head start, it received an additional infra red sensor for distance measurement. The competition starts when predator and prey are exposed to a random, unknown environment bordered by walls. The predator has to catch the prey while the prey has to escape to a hideout. The default robot shape (morphology) provided was not fixed. Every group was encouraged to change their robots default shapes to their needs and strategies. They could relocate sensors, body material could be removed or additional material could be attached. While designing this robotic platform, the focus was on using cheap, commercially available parts and open-source software. For more detailed information about the robot's hardware and software, the competition's setup and rules see (EmbedIT, Assaf et al. 2010). The total material costs of one robot is about USD 300. The reason why we chose a custom made platform and introduced the teachers to the Arduino board was the fact that most of the teachers were already familiar with the LEGO NXT platform. On the contrary to the LEGO NXT platform which is very neat in its design the custom made robot looks like a 'real' robot (one can see the electronic components, cables, batteries etc.). We deliberately wanted them to see what's inside a robot.



**Figure 2:** Predator and prey robots. The predator has to catch the prey, whereas the prey has to escape and look for the hideout. Both robots are equipped with odometry, ultra sonic, light and touch sensors. The prey has an additional infra red sensor for distance measurement.

## Results

The teachers enjoyed both robot competitions very much. They were especially motivated to work on the predator & prey robot competition project. Some examples of predator and prey robots built by the teachers are shown in Figure 3. Most of the groups modified the default robot configuration completely. Sensors were relocated, additional features such as tentacles, bumpers, cases or covers for the light or ultra sonic sensors were added by using materials such as LEGO blocks, wood, plastic and metal. The performance of the robots differed but the range was not too big. However, some groups came up with sophisticated control strategies whereas others only managed to implement some basic obstacle avoidance behavior. Interestingly, there was no correlation between the background of the teachers and the performance of their robots. Some teachers from non-technical backgrounds were among the ones with the best results most likely due to their motivation and general interest in this topic.



**Figure 3:** A selection of different predator and prey robots built by the teachers. The robots were extended with LEGO blocks, wood, plastic and metal parts. The teachers built additional features such as tentacles, bumpers, cases or filter covers for the sensors.

The feedback of the first robotics class in 2010 was received mostly on personal discussions with the teachers and was overall very positive. It has to be mentioned that these particular participants were extraordinary in terms of programming skills and motivation. The data described here has been collected using questionnaires in the second class in 2011. Even though the number of answers is rather low for a meaningful quantitative evaluation, the results of this small survey reflect the overall experiences we made throughout both years. Table 1 shows the percentage of affirmative replies to questions related to the use of robots in class and the robotic platform chosen. Interestingly, the majority appreciated the use of a custom made robotic platform (78.6%) and the fact that they had to build the robot's body on their own (86.7%). At the same time an even more significant 84.6% would have liked to use LEGO NXT for the predator & prey competition. All teachers would consider to use robots for their own informatics class (100%). Despite our impression that for those teachers with mathematical background programming was less difficult, only a small number of teachers found it necessary to have a profound knowledge of Mathematics for such a robotics class (15.4%).

Question	% of affirmative replies	Number of teachers
Did you like to work with a custom robotic platform based on the Arduino board?	78.60%	14
Did you like the assignment to modify the robot's body?	86.70%	15

Would you prefer to use LEGO NXT for the predator & prey project?	84.60%	13
Will you consider to use robots for your own informatics class?	100.00%	14
Do you think such a robotics class requires profound knowledge of Mathematics?	15.40%	13

**Table 1:** Percentage of positive answers to questions concerning the use of robots in class and the robotic platform chosen.

Table 2 shows the results of questions regarding their general impression about the class and the predator & prey project where '1' represents 'not at all' and '10' stands for 'very much'. To the question whether they have learned more about robotics the answer is high namely, '8'. The difficulties of the predator & prey project and the respective programming task were estimated as high as well ('8' for the former and '7' for the latter). At the same time with an average of '8' the fun factor was concluded to be very high.

Question	1 = not at all 10 = very much	Number of teachers
Have you learned more about robotics?	8	16
Difficulty of the predator & prey project	8	15
Difficulty of programming	7	16
Fun factor	8	16

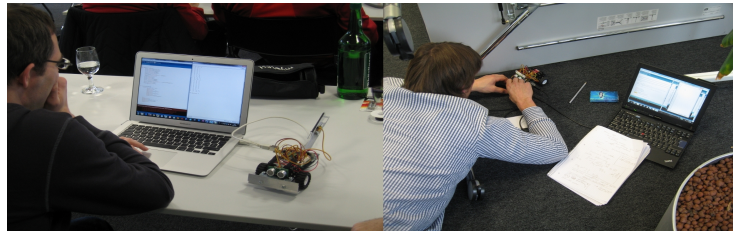
**Table 2:** General impression about the class and the predator & prey project.

## Discussion

The teachers appreciated the hands-on approach of this class and the fact that they could work on a practical project rather than having to write a theoretical exam by the end of the semester. Additionally, they enjoyed the group work and the competition as a game. Many of them had the feeling to have learned a lot about robots and felt confident enough to be able to evaluate robot kits in the future. The opinions were very diverging about the custom made robotic platform. On one hand the teachers appreciated the opportunity to be introduced to an alternative platform (Arduino) next to LEGO NXT. On the other hand they would have preferred to work with LEGO NXT since most of the schools already possess LEGO NXT kits due to their robustness and popularity. The transfer from this robotics class to their school activities would thus be easier. As for the LEGO competition described earlier in this paper, they appreciated the fact that they could directly apply it in their class. However, the aim of this robotics class was more focused on teaching robotics rather than providing ready-made exercises for class use. Many teachers pointed out the suitability of the Arduino platform for individual student projects whereas LEGO NXT would be more adequate for general use in class due to its robustness.

In this survey all participants considered using robots for their own informatics class. They argued that students respond well to hands-on exercises. They pointed out the importance of building and construction skills which are neglected by the young people more and more nowadays. Robots were seen as suitable especially for teaching programming and logical reasoning, since the effect of edited code can be tested and observed directly on the robot.

The fact that mathematical knowledge is, in their opinion, not absolutely necessary for attending a robotics class supports the aim to target students who have broader interests rather than only in Mathematics. For the predator & prey project it has to be mentioned that in hindsight the assignment goals were too ambitious (catching the prey, escaping). The teachers were not aware of the fact that working with robots can be very problematic due to noise of sensors, inaccurate motors etc. These experiences that are on one hand important and valuable can on the other hand easily lead to frustration. It is therefore important not to aim too ambitiously and rather to increase complexity step by step. The high fun factor also shows that for teachers a hands-on approach is fun and a robot competition can lead to a high level of engagement (Figure 4).



**Figure 4:** The high fun factor shows that a hands-on approach is fun and a robot competition can lead to a high level of engagement.

## Conclusion

Overall, the robotics class for teachers was a success. The feedback was positive and the educational objectives were met. Most of the teachers saw robots as a suitable tool for teaching informatics. Hence, the approach to use a custom robotics platform for teacher education was found to be suitable. However, for future teaching activities we will point out that fun and frustration go hand in hand while working with robots. Therefore, assignments should not be designed too ambitiously in order to ensure motivation through the feeling of accomplishment.

## References

- Alimisis, D., Moro, M., Arlegui, J., Frangou, S., Papanikolaou, K. (2007). Robotics & Constructivism in Education: the TERECoP project. In *Proceedings of EuroLogo 2007*, Bratislava
- Arduino. The Arduino project. <http://www.arduino.cc/>
- Assaf, D., Pfeifer, R. (2010). Robot Competition with Teachers. In *Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010)*, pp. 514-521, ISBN 978-3-00-032863-3
- Balogh, R. (2010). Acrob – an Educational Robotic Platform. *AT&P Journal Plus 2, Robotics in Education*, pp. 6-9, ISSN 1336-5010, (2010)
- Bredenfeld, A., Leimbach, T. (2010). The Roberta Initiative. In *Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010)*, pp. 558-567, ISBN 978-3-00-032863-3
- Bredenfeld, A., Hofmann, A., Steinbauer, G. (2010). Robotics in Education Initiatives in Europe – Status, Shortcomings and Open Questions. In *Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010)*, pp. 568-574, ISBN 978-3-00-032863-3
- EmbedIT. <http://www.embed-it.ch>
- Kabátová, M., Pekárová, J. (2010). Lessons learnt with LEGO Mindstorms: from beginner to teaching robotics. *AT&P Journal Plus 2, Robotics in Education*, pp. 51-56, ISSN 1336-5010
- Menegatti, E., Moro, M. (2010). Educational Robotics from high-school to Master of Science. In *Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010)*, pp. 639-648, ISBN 978-3-00-032863-3
- Miller, J., Ward, B., Sienkiewicz, F., Antonucci, P. (2010). ITEAMS: an Out-Of-School Time Program to Promote Gain in Fundamental Science Content and Enhance Interest in STEM Careers for Middle School Students. In: *Proceedings of The 8<sup>th</sup> Intl. Conf. on Education and Information Systems, Technologies and Applications (EISTA 2010)*
- Pfeifer, R., Bongard, J. (2006). *How the Body Shapes the Way We Think – A New View of Intelligence*. MIT Press, ISBN 978-0-262-16239-5
- Piaget, J. (1952). *The origins of intelligence in children*. New York: International Universities Press.
- Piaget, J. (1954). *The construction of reality in the child*. New York: Basic Books.
- Rinderknecht, M., Ruesch, J., Hadorn, M. (2007) The Lagging Legs Exploiting Body Dynamics to Steer a Quadrupedal Agent. In *proceedings of the International Conference on Morphological Computation (ICMC)*, Venice, Italy, March 2007
- Rocard, M. et al. (2007). *Science Education now: a renewed pedagogy for the future of Europe*. EUR22845, ISBN 978-92-79-05659-8, European Communities, Brussels
- Sardone, N. (2010). Developing Information Technology (IT) Fluency in College Students: An Investigation of Learning Environments and Learner Characteristics. In: *Proceedings of The 8<sup>th</sup> Intl. Conf. on Education and Information Systems, Technologies and Applications (EISTA 2010)*





# Embedit – an Open Robotic Kit for Education

Reprinted from:

Assaf, D. and Pfeifer, R. (2011). *Robotics as Part of an Informatics Degree Program for Teachers*, In Research and Education in Robotics – EUROBOT 2011, Communications in Computer and Information Science, volume 161, pages 29–39. Springer.

## EmbedIT - an Open Robotic Kit for Education

Dorit Assaf and Rolf Pfeifer

Artificial Intelligence Laboratory  
Department of Informatics  
University of Zurich  
Switzerland  
`{assaf,pfeifer}@ifi.uzh.ch`

**Abstract.** Robots have often been used as an educational tool in class to introduce kids to science and technology, disciplines that are affected by decreasing enrollments in universities. Consequently, many robotic kits are available off-the-shelf. Even though many of these platforms are easy to use, they focus on a classical top-down engineering approach. Additionally, they often require advanced programming skills. In this paper we introduce an open robotic kit for education (EmbedIT) which currently is under development. Unlike common robot kits EmbedIT enables students to access the technical world in a non-engineering focused way. Through a graphical user interface students can easily build and control robots. We believe that once fascination and a basic understanding of technology has been established, the barrier to learn more advanced topics such as programming and electronics is lowered. Further we describe the hardware and software of EmbedIT, the current state of implementation, and possible applications.

**Key words:** Robotic Kits, Educational Robotics, Edutainment, Embedded Systems, Rapid Prototyping in Robotics

### 1 Introduction

The number of students showing interest in pursuing a career in Science, Technology, Engineering or Math is decreasing in the USA [1] as well as in Europe [2]. In order to maintain economical competitiveness, education in these disciplines is crucial. This phenomenon has gained a lot of attention from teachers, researchers, politicians, authorities and other stakeholders. Consequently, a vast number of initiatives have been developed to improve the situation. A popular educational approach to motivate young people in learning is based on the constructivist/constructionist paradigm, where learning through play can contribute to the construction of knowledge [3][4]. Active learning environments through the use of interactive lessons, friendly competitions, and trial and error are therefore preferred. Robots have been used in the last decade to introduce kids and especially girls [5] to science and technology [6]. Class activities with robots range

from kindergarten to high secondary school. A large number of robot competitions emerged such as the FIRST Lego League or RoboCupJunior, all with the aim to engage young people in these disciplines [7]. Consequently, many robot kits have been developed in research projects as well as in commercial companies. A popular robot kit used in educational robotics courses is the LEGO NXT [8]. Even though this platform is very easy to use, it focuses on a classical top-down engineering approach, where the actual robot has to be planned, built out of predefined parts and programmed using classical programming routines. Many other educational robotic platforms require advanced programming skills in C/C++.

In this paper we describe an open robotic kit (EmbedIT) that we are currently developing. The aim is to provide an open-source platform that enables teachers and students to easily build and control robots. In addition, our robot kit can be used as a rapid prototyping tool for researchers. With EmbedIT we try to overcome the constraints of classical engineering focused toolkits. We believe since young people are used to user-friendly applications in their everyday life, we should incorporate such interfaces in our kit. The user should be able to play with sensors and actuators without having to deal with the technical details. Furthermore, he should be able to build and implement a robot within a short time, and consequently, while testing and revising it, getting a deeper understanding about these sensors and actuators. We believe that once fascination and a basic understanding of technology has been established, motivation is increased to get familiar with more advanced topics such as programming and electronics. However, we don't solely target one group of users. EmbedIT is completely open-source and therefore technically more skilled users can reprogram and adapt the platform as they wish.

In the rest of this paper, besides the related work, the EmbedIT's hardware and software is described. This is followed by application examples, discussion, conclusion and future work.

## 2 Related Work

The field of educational robotics is growing due to the fact that robots are a proven effective learning tool [7]. A widely used robotic platform for educational robotics is the LEGO NXT [9][10]. It provides actuators, a variety of sensors, building blocks as well as an easy-to-use graphical programming language. Additionally, the LEGO NXT platform can be programmed using higher programming languages, such as JAVA.

A low cost educational robotic platform is the Asuro robot [11]. By soldering all electronic components to the PCB the user has to assemble the robot from scratch. Asuro is designed to be a wheeled robot, thus the user has no much flexibility to modify the default shape. It is programmed with C/C++ programming language. Other educational robotic platforms [12] use the popular Arduino boards [13]. We also used a small custom made wheeled robot based on the Arduino board to teach robotics to secondary school teachers [14]. Arduino

is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. Furthermore, the Arduino project provides custom C/C++ libraries to program the on board microcontroller. Hardware design and software is open-source and a large community grew around the platform contributing libraries, tutorials, hardware designs etc. Meanwhile, several Arduino control board designs have been developed, each suited for different applications. However, even though there are plenty of well documented online tutorials, skills in electronics are needed in order to attach sensors and actuators to these boards (which have to be purchased separately). The Arduino boards can be damaged easily by connecting additional components in the wrong way.

A fairly new project, but very related to the EmbedIT platform is .NET Gadgeteer of Microsoft research [15]. It is a rapid prototyping platform for small electronic gadgets and embedded hardware devices. It combines object-oriented programming and solderless assembly of electronics using a kit of hardware modules. Unlike the Arduino platform .NET Gadgeteer provides all kinds of sensors and actuators that can be plugged into the main board. This prevents problems caused by wrong connections.

Phidgets offers similar to .NET Gadgeteer a variety of sensors and motor controller modules which are all connected through an USB interface to the computer [16]. Additionally, the Phidgets libraries support the most common programming languages.

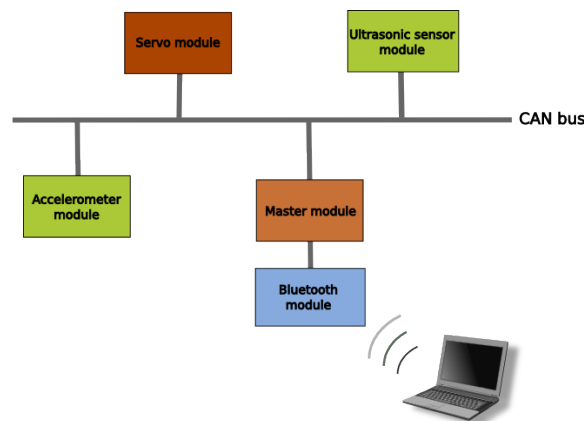
Nevertheless, Asuro, Arduino, .NET Gadgeteer as well as Phidgets require advanced programming skills.

### 3 EmbedIT - An Open Robotic Kit

Currently we are implementing the robot kit's hardware (electronics) and software. Building blocks such as structural material, wheels etc. will not be provided by EmbedIT. However, it is planned to provide design solutions to interface EmbedIT's electronics together with recommended motors and sensors with off-the-shelf mechanical construction kits such as LEGO blocks [17], Stokys [18], Meccano [19] etc.

The components of the EmbedIT platform are hardware modules consisting of small embedded systems (individual printed circuit boards equipped with an Atmel AVR microcontroller). Each module is an independent entity that serves either a control, sensory, actuation or communication purpose. According to a module's purpose a specific control program is running on its microcontroller. Additionally, each module carries its own unique identification number. A CAN data bus (two conductor cable) is the connecting piece between the modules. Each module can easily be attached and detached from the bus. The master module, which is a control module, controls the communication traffic on the bus. In addition, the master module serves as an interface to the user. A Java application, the EmbedITApp, running on the user's PC connects to the master module through a communication module, that can either be a Bluetooth, USB

or serial (RS232) interface. In Fig. 1 the master module is attached to the CAN bus and a Bluetooth module is attached to the master module (communication modules are not attached to the bus directly, they have to be attached to a module). The EmbedITApp connects to the master module through the Bluetooth communication module. Additionally in Fig 1, one actuation module (servo) and two sensor modules (accelerometer, ultrasonic sensor) are connected to the bus.

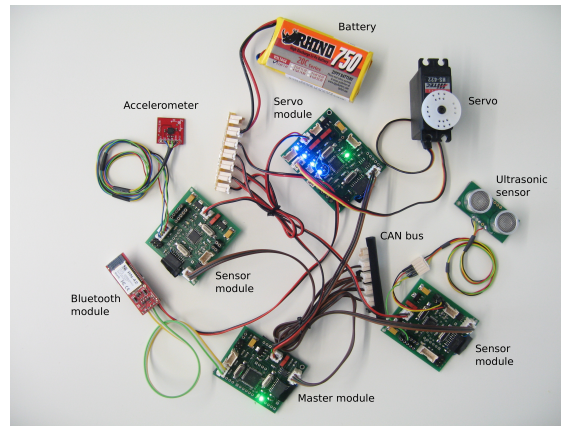


**Fig. 1.** A master module is attached to the CAN bus and a Bluetooth module is attached to the master module. The EmbedITApp running on the PC connects to the master module through the Bluetooth communication module. Additionally, a servo module, an accelerometer module and an ultrasonic sensor module are connected to the bus.

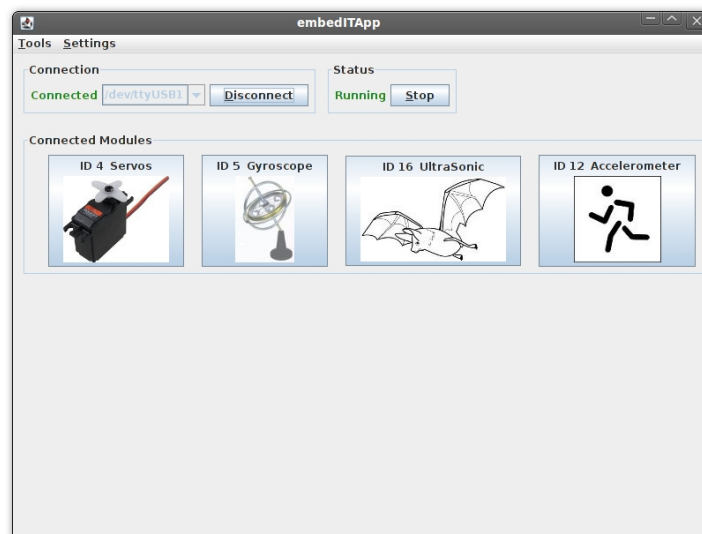
Fig. 2 shows the real modules, a Bluetooth module, a master module, two sensor modules (accelerometer, ultrasonic), the CAN bus, a power supply and a servo module with one servo attached. Once a connection between the EmbedITApp and the master module has been established, the master module returns the ID and module type (sensor type, actuator type) of all hardware modules that are physically connected to the bus.

Fig. 3 shows an example of the EmbedITApp where a servo module and three sensor modules (gyroscope, ultrasonic, accelerometer) are listed. Each available module is represented with its ID, icon and a button within the application. While clicking on a module's button, its control panel opens.

Fig. 4 shows the control panel of the servo module. The servo module is able to control four servos at the same time. From this control panel commands such as connect/disconnect servos or move to discrete servo positions can be sent. Another possibility is to let the servo motors change positions in a continuous, sinusoidal way (alternating from 0 to 180). Different sine parameters such as amplitude (servo angle range), offset (shift of the servo middle axis), phase lag

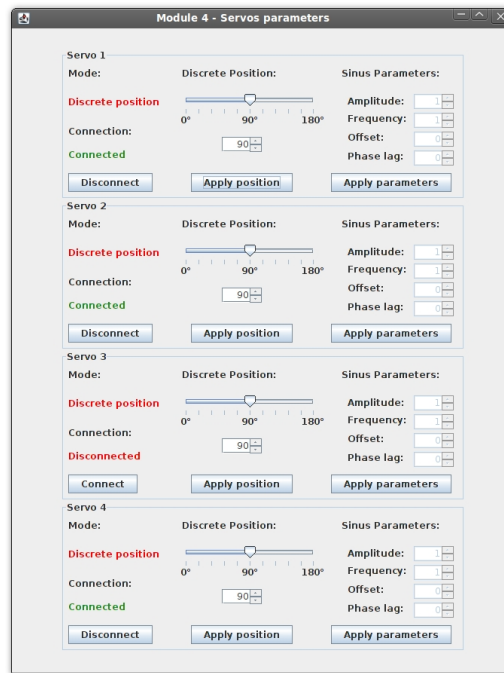


**Fig. 2.** Some EmbedIT modules: A Bluetooth module, a master module, two sensor modules (accelerometer, ultrasonic), the CAN bus, a power supply and a servo module with one servo attached.



**Fig. 3.** An example of the EmbedITApp where a servo module and three sensor modules (gyroscope, ultrasonic, accelerometer) are listed. Each available module in the application is represented with its ID, icon and a button.

(synchronization of all servos) and frequency can be applied. The EmbedITApp sends the commands to the master module which forwards them to the target module on the bus. Servo positions can be changed while dragging the servo position sliders. After pushing the apply buttons on the control panel, the servo positions are changed physically.



**Fig. 4.** The control panel of the servo module. The servo module is able to control four servos at the same time. From this control panel commands such as connect/disconnect servos, move to specific servo position as well as continuous position change in a sinusoidal way can be applied.

For each module on the bus a control panel can be opened where module specific commands and settings can be applied. The master module manages the requests coming from the EmbedITApp and forwards them to the specific hardware modules. These targeted hardware modules execute the requests (motor position, get sensor data) and reply to the master module which acknowledges the EmbedITApp's commands. To achieve robustness the EmbedITApp resends the request three times in case of unacknowledged commands. To minimize traffic on the CAN bus only high level commands are being sent to the individual modules (e.g. start, stop motors etc.). Since each module has module specific

algorithms implemented computation is reduced for the master module. Through the plug and play nature of the hardware modules on the bus as well as the possibility to acquire sensor data and to control modules through a GUI, usability increases significantly. With other platforms a user has to program a lot of code in order to be able to do similar tasks. At this state the user is able to remote control and monitor the hardware. By means of a graphical programming environment which is currently under development the user will be able to define the control of sensors and actuators such that the hardware will be able to run autonomously. It has to be mentioned that the preprogrammed source code of each EmbedIT module is not hidden. A more experienced user can easily change code (C/C++) and upload it to the modules.

EmbedIT's hardware designs and source codes are completely open and will be free available on the project website [20] with the release of the stable version. Inspired by the Arduino project, we would like to enable a community to extend and adapt the hardware and software once it is released. The main goal is to encourage users from all kinds of backgrounds to use this kit. By means of the EmbedITApp users with no background in electronics or programming are able to access and control the modules. In addition, more advanced users have the possibility to extend the source code by themselves. Since the hardware and software interfaces are open and common standards are used, it is possible to interface the modules with other kits or projects, for example with an Arduino board. By adding the required CAN components to the Arduino board and including our CAN library on the software side, the board is easily connected with EmbedIT.

## 4 Application Examples

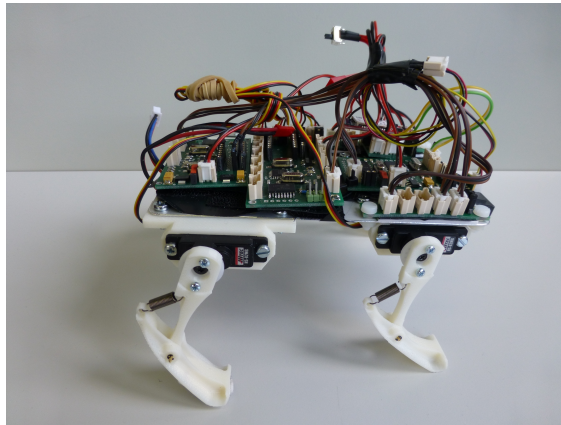
As we are currently developing the hardware and software of EmbedIT, more modules are being added to the kit. Currently in addition to the servo module common sensor modules exist such as light sensors, ultrasonic sensors, accelerometers, potentiometers, gyroscopes. The goal is to have a repertoire of many different kinds of sensors and actuators, also unusual ones such as shape memory alloy, whisker sensors [21] etc. There are many possible applications for this kit and they are not limited to education only. Other uses could be e.g. a rapid prototyping tool for researchers as well as a playground for hobbyists. In this section we describe application examples where two different robots (a quadrupedal running robot and a skating robot) are controlled by the same EmbedIT modules. Both robots use in addition to the master and Bluetooth modules a servo module that controls four servos in a sinusoidal positioning mode. Each robot requires its specific sine parameters that had to be tuned manually by trial and error using the EmbedITApp interface. These two examples show how easy and with no programming knowledge one can already control a robot. Furthermore, the EmbedIT control interface eases the exploration of control parameters in cases where this kind of approach is required.



#### 4.1 A quadrupedal running robot controlled with EmbedIT

Iida et al. [22] applied a parsimonious control strategy for a running quadrupedal robot. The motors were controlled by a simple oscillatory position control without additional sensory feedback. The robot had four identical legs each of which consisted of one servo motor and a series of two limbs connected through a passive elastic joint.

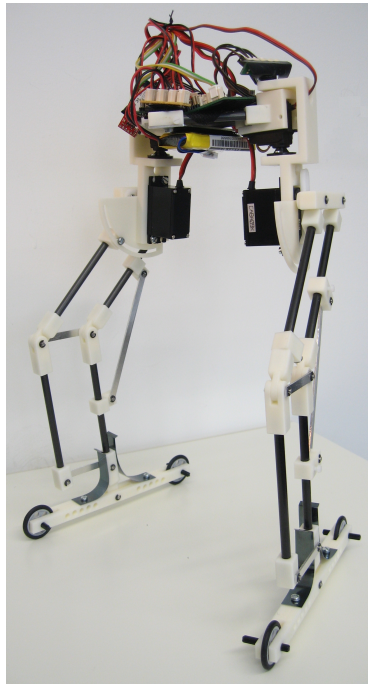
We rebuilt a similar puppy robot based on the same concept and controlled by EmbedIT modules (Fig. 5). By means of the EmbedITApp's servo control interface different sine parameters could be applied to the robot and therefore different running gaits could be achieved.



**Fig. 5.** A quadrupedal running robot controlled with EmbedIT modules (a master module, a Bluetooth module, a servo module, a sensor module, a battery). The robot has four identical legs each of which consists of one servo motor and a series of two limbs connected through a passive elastic joint.

#### 4.2 A skating robot controlled with EmbedIT

The same EmbedIT modules used for the quadrupedal running robot (Fig. 5) were reused to control a skating robot (Fig. 6). The two servos on the hips control the bending and stretching movement of the legs and the two other servos above the hips adjust the angle between the feet. The goal was to control the robot such that it achieves a forward skating movement while alternately relocating its center of mass from one leg to the other (that, in a simplified way, is what skaters actually do). Therefore discrete positioning has been applied to the feet angle servos while the hip servos were controlled in a sinusoidal positioning mode.



**Fig. 6.** A skating robot controlled with EmbedIT modules (a master module, a Bluetooth module, a servo module, a battery). The two servos on the hips control the bending and stretching movement of the legs and the two other servos above the hips adjust the angle between the feet.

## 5 Discussion

EmbedIT differs from other robotic kits with respect to usability and user-friendliness. Our goal is to enable users from different kinds of backgrounds and interests to explore the world of technology while accessing it in a creative, playful and non-engineering focused way. We believe that building own robots result in a basic understanding about how they work. This can lower the barrier to move on to more advanced topics of engineering such as programming and electronics which can be explored with the kit as well.

EmbedIT is contributing to education in such a way that it provides an alternative to common robot kits on the market. The goal is the same - teaching technology - but the approach is different. The user-friendliness of EmbedIT can encourage teachers to incorporate it in class as a learning tool. We intentionally chose EmbedIT to be open source to encourage the growth of a contributing community. This could consequently lead to the availability of cheap hardware from which schools can profit as well.

## 6 Conclusion and Future Work

In this paper we referred to the problem of decreasing enrollments in science and technology disciplines in Universities. The field of educational robotics is growing due to the fact that robots are a proven effective learning tool for education in these disciplines. Consequently, many robot kits have been developed in research projects as well as in commercial companies. However, most of them focus on a classical top-down engineering approach, where the actual robot has to be planned, built out of predefined parts and programmed using programming languages such as C/C++. We are developing EmbedIT, an open robotic kit that enables users to build robots in an easy and user-friendly way. This can be highly motivating and may lower the barrier to move on to more advanced topics in engineering.

Right now we are developing, testing and revising the hardware and software of EmbedIT. More modules have to be added to the repository. At the moment servos can be controlled through the EmbedITApp interface and sensor data can be read. To combine sensor data and servo commands a graphical user interface has to be implemented. In addition, interface solutions for attaching EmbedIT's modules to mechanical structures (for instance by using mechanical toolkits) have to be developed. Since we are interested in including the end-users feedback in our design process, it is important to communicate and publish the project idea as well as to test the kit with students at an early stage. In order to evaluate the approach workshops using EmbedIT in upper secondary schools are planned. A survey will be conducted to investigate the impact EmbedIT produces upon students, regarding change of attitude and self-confidence towards studying technology. Once the EmbedIT hardware and software achieves a first stable release state, it will be available under an open-source license on the project website [20].

**Acknowledgments.** We would like to thank Wolfgang Pils and Andreas Bertschi for the idea and the design of the skating robot (Fig. 6).

## References

1. Miller, J., Ward, B., Sienkiewicz, F., Antonucci, P.: ITEAMS: an Out-Of-School Time Program to Promote Gain in Fundamental Science Content and Enhance Interest in STEM Careers for Middle School Students. In: Proceedings of The 8th Intl. Conf. on Education and Information Systems, Technologies and Applications (EISTA 2010)
2. Rocard, M. et al.: Science Education now: a renewed pedagogy for the future of Europe. EUR22845, ISBN 978-92-79-05659-8, European Communities, Brussels (2007)
3. Piaget, J.: The origins of intelligence in children. New York: International Universities Press. (1952)
4. Piaget, J.: The construction of reality in the child. New York: Basic Books. (1954)
5. Bredenfeld, A., Leimbach, T.: The Roberta Initiative. In Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAN 2010), pp. 558-567, ISBN 978-3-00-032863-3
6. Alimisis, D., Moro, M., Arlegui, J., Frangou, S., Papanikolaou, K.: Robotics & Constructivism in Education: the TERECoP project. In Proceedings of EuroLogo 2007, Bratislava, (2007)
7. Bredenfeld, A., Hofmann, A., Steinbauer, G.: Robotics in Education Initiatives in Europe Status, Shortcomings and Open Questions. In Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAN 2010), pp. 568-574, ISBN 978-3-00-032863-3
8. LEGO NXT: <http://mindstorms.lego.com>
9. Kabatova, M., Pekarova, J.: Lessons learnt with LEGO Mindstorms: from beginner to teaching robotics. AT&P Journal Plus 2, Robotics in Education, pp. 51-56, ISSN 1336-5010, (2010)
10. Menegatti, E., Moro, M.: Educational Robotics from high-school to Master of Science. In Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAN 2010), pp. 639-648, ISBN 978-3-00-032863-3
11. ASURO robot: [http://www.arexx.com/arexx.php?cmd=goto\&cpam=p\\_asuro](http://www.arexx.com/arexx.php?cmd=goto\&cpam=p_asuro)
12. Balogh, R.: Acrob – an Educational Robotic Platform. AT&P Journal Plus 2, Robotics in Education, pp. 6-9, ISSN 1336-5010, (2010)
13. The Arduino project. <http://www.arduino.cc/>
14. Assaf, D., Pfeifer, R.: (2011). Robotics as Part of an Informatics Degree Program for Teachers. In Proceedings of Society for Information Technology & Teacher Education International Conference 2011 (pp. 3128-3133). Chesapeake, VA: AACE.
15. The Gadgeteer project. <http://research.microsoft.com/en-us/projects/gadgeteer/>
16. Phidgets. <http://www.phidgets.com/>
17. LEGO: <http://www.lego.com>
18. Stokys: <http://www.stokys.ch/>
19. Meccano: <http://www.meccano.com>
20. The EmbedIT project. <http://www.embed-it.ch>
21. Lungarella, M., Hafner, V., Pfeifer, R., Yokoi, H.: Artificial Whisker Sensors in Robotics. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), (2002)

12      EmbedIT - an Open Robotic Kit for Education

22. Iida, F., Gomez, G., Pfeifer, R.: Exploiting body dynamics for controlling a running quadruped robot In Proceedings of the 12th International Conference on Advanced Robotics. ICAR '05 (2005)



# Extending Mechanical Construction Kits to Incorporate Passive and Compliant Elements for Educational Robotics

Reprinted from:

Assaf, D. and Larsen, J. C. and Reichardt, M.(2012) *Extending mechanical construction kits to incorporate passive and compliant elements for educational robotics.*, In Proceedings of 3rd International Conference on Robotics in Education (RiE 2012), pages 33–40.

# Extending Mechanical Construction Kits to Incorporate Passive and Compliant Elements for Educational Robotics

Dorit Assaf  
Artificial Intelligence Laboratory  
Department of Informatics  
University of Zurich  
Zurich, Switzerland  
Email: assaf@ifi.uzh.ch

Jørgen Christian Larsen  
Modular Robotics Research Lab  
The Maersk Mc-Kinney Moller Institute  
University of Southern Denmark  
Odense, Denmark  
Email: jcla@mml.sdu.dk

Markus Reichardt  
Reichardt Informatik AG  
Höhenweg 22  
Münchenstein, Switzerland  
Email: markus@reichardt.ch

**Abstract**—Robots are a popular educational tool to introduce science, technology, and engineering to students. The field of educational robotics is growing and consequently a number of educational robot kits have been developed within the last decade. Our laboratories have a tradition of teaching embodied artificial intelligence and biomechanics to students with different backgrounds. The robots we use both for research and education are usually built incorporating compliant materials as well as passive dynamics. These kind of properties are often not available in classical robot kits or mechanical construction kits. In this paper we describe some of the robots we use for education. So far we built the robots using 3D printing technology which is convenient but too expensive for class use. Our aim is to find cheaper, commercially available solutions. After a short review on educational robot kits and mechanical construction kits we describe interface solutions between several kits. Further we show some solutions to incorporate compliant materials and passive dynamics to traditional mechanical construction kits by using cheap and widely available materials.

## I. INTRODUCTION

The notion of embodiment, which has formed the major research target of the Artificial Intelligence Laboratory (AILAB) over the last 15 years, has dramatic implications for our understanding of intelligence [1]. For example, behavior is not the result of brain processes only, but of a subtle interplay between brain, body (morphology and materials) and environment; an insight that contradicts the classical Cartesian position. According to the embodied artificial intelligence perspective, morphological and material characteristics of an organism can take over a large part of its functionality [2]. We use the term morphological computation to designate the fact that some of the control or computation can be taken over by the dynamic interaction derived from morphological properties (e.g. the passive forward swing of the leg in walking, the spring-like properties of the muscles, and the weight distribution) [3]. By taking morphological computation into account, an agent will be able to achieve not only faster, more robust, and more energy-efficient behavior, but also more situated exploration by the agent for the comprehensive understanding of the environment.

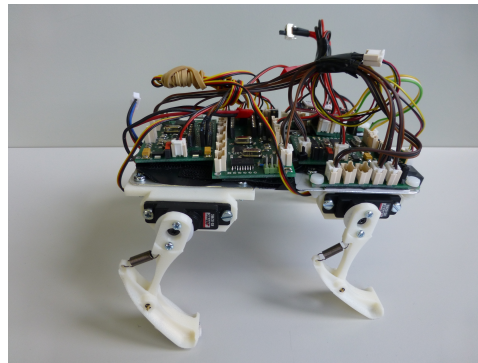


Fig. 1. A puppy robot used for educational purposes.

Our laboratories (AILAB, Modular Robotics Research Lab) have a tradition in teaching the principles of embodied intelligence to students with different backgrounds. For instance in the context of an informatics degree program for high school teachers, we (AILAB) conducted a LEGO NXT robot competition where solely the morphology was allowed to be changed in order to achieve faster locomotion [14]. The initial LEGO robot morphology has been inspired by a robot built by Rinderknecht et al. [5]. In a variety of other teaching activities we used robots that locomote using passive dynamics (Fig. 1), inspired by the quadruped robot of Iida et al. [4] as well as unusual robots inspired from both research and arts. We used for instance a smaller version of the RHex robot [6] (Fig. 2) and an actuated one of Theo Jansen's Strandbeest<sup>1</sup> (Fig. 3) for several robot workshops.

With the exception of the LEGO NXT robot competition example mentioned above we usually use our open toolkit "EmbedIT" for the robot control (electronics and software)[7].

<sup>1</sup><http://www.strandbeest.com/>



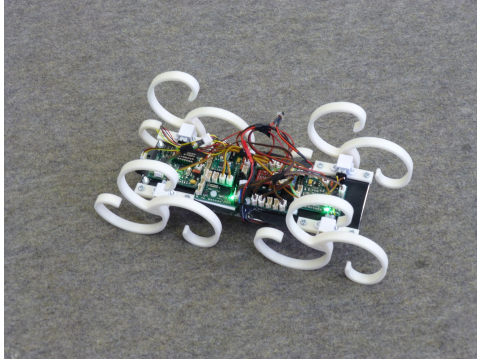


Fig. 2. A smaller version of the RHex robot used for educational purposes.

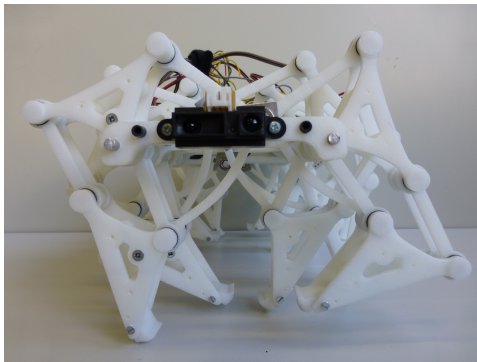


Fig. 3. An actuated version of Theo Jansen's Strandbeest used for educational purposes.

For the mechanical construction we custom built the robots in the past, using 3D printing technology (all the white plastic parts in Fig. 1,2,3). The possibility to 3D print the desired parts is convenient and fast. They are lightweight, high in precision such that generally no additional machining is necessary. The parts are further surprisingly stable, considering the strong impact forces that act especially on RHex's wheels and Puppy's legs. Even after several classes not a single 3D printed part had to be replaced (opposed to the motors which frequently broke due to jammed gears). However, 3D printing is still expensive, not particularly environmental friendly and not necessary if reasonable alternatives are available. Additionally, many institutions don't have 3D printing infrastructure, the required software licenses and knowledge to design parts using CAD. In trying to solve this, a trend is emerging towards low-cost personal fabrication solutions with projects such as RepRap<sup>2</sup>, fab@home<sup>3</sup> or MakerBot<sup>4</sup>. However, if the robot

<sup>2</sup><http://reprap.org/wiki/RepRap>

<sup>3</sup><http://www.fabathome.org/>

<sup>4</sup><http://www.makerbot.com/>

parts are not too specific and complicated, a cheap off-the-shelf solution of mechanical construction components is still preferable, especially if they are made out of reusable, stable and lightweight material such as aluminum.

Building objects (cars, trucks, planes etc.) using mechanical construction kits had been very popular at the beginning of the last century. Brands such as "Meccano" are widely known in the generation born in the 1940's. These kind of playing activities are no longer popular with young people and thus traditional manufacturers such as Meccano, Märklin and Stokys suffered.

This paper describes our search for a low-cost solution to build robots with unusual shapes using compliant, passive dynamic elements for educational purposes. We give a short review on off-the-shelf robot kits and mechanical construction kits in order to identify their advantages and disadvantages. Since most of the classical mechanical construction kits do not support any interfaces to standard actuators we show some easy solutions how to overcome this constraint. We describe how to use common and cheap materials everyone can find at home or in a conventional do-it-yourself store to build unusual robots and without the need of 3D printing technology. We show some examples how to interface proprietary robot kits with other construction kits to achieve a greater construction flexibility. Further we introduce our robotic construction kit "LocoKit", which is currently under development [15]. This system is targeted towards legged robots, and promises to make it possible to build dynamically walking robots in a fast and easy way. The LocoKit is described more deeply in section V.

## II. A REVIEW ON ROBOT KITS AND MECHANICAL CONSTRUCTION KITS FOR EDUCATION

In the following section we list a number of robot platforms that are usually used for educational robotics and robot competitions. The list is far from complete, however, the robots mentioned are a good representation of what is usually used. This is followed by a short review on mechanical construction kits. Also here we give a broad overview of different kits using different materials and concepts how to connect the elements together.

From this short survey we select one or two example platforms and describe how to interface them with each other and how to extend them using other materials, which do not originally belong to the toolkit in order to build the robots we would like to use in class.

### A. Robot Kits

Robots have been used in the last decade to introduce kids to science and technology [8],[9]. Class activities with robots range from kindergarten over secondary school to universities. A large number of robot competitions emerged such as the FIRST Lego League, Eurobot, RoboCupJunior, Botball or Robolympics, all with the aim to engage young people in these disciplines [10]. Consequently, many robot kits have been developed in research projects as well as in commercial

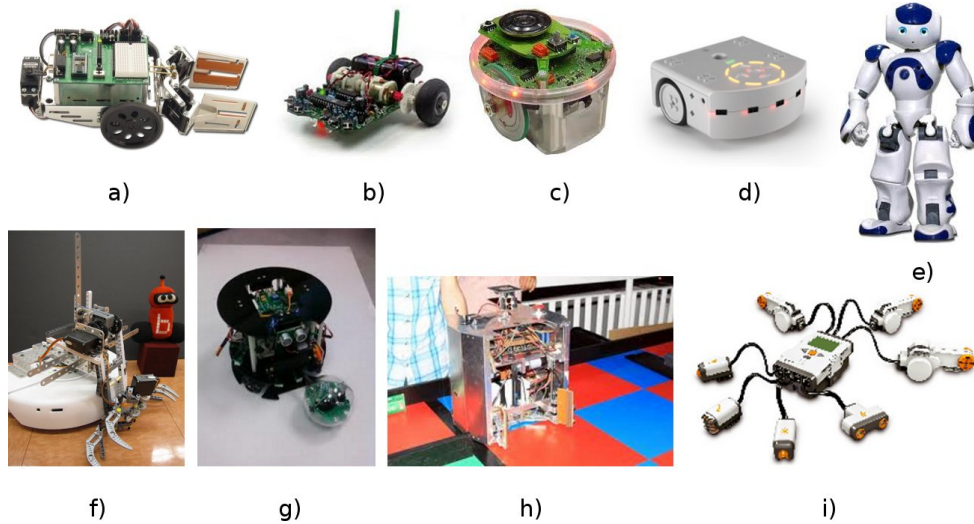


Fig. 4. A collection of robots used in education and competitions. a) boe-bot, b) Asuro, c) e-puck, d) ThymioII, e) Nao, f) Botball, g) a robot used in RoboCup, h) a robot used in Eurobot, i) LEGO NXT.

companies. A widely used robotic platform for educational robotics is the LEGO NXT<sup>5</sup> (Fig. 4i) [11],[12]. It provides actuators, a variety of sensors, building blocks as well as an easy-to-use graphical programming language. Additionally, the LEGO NXT platform can be programmed using high-level programming languages, such as JAVA. A low cost educational robotic platform is the Asuro<sup>6</sup> (Fig. 4b). By soldering all electronic components to the PCB the user has to assemble the robot from scratch. Asuro is designed to be a wheeled robot, thus the user has not much flexibility to modify the default shape. Many other educational robotic platforms use the popular Arduino<sup>7</sup> boards [13]. We also used a small custom made wheeled robot based on the Arduino board to teach robotics to secondary school teachers [14]. Other commercial robot platforms designed for educational purposes are E-puck<sup>8</sup> (Fig. 4c), ThymioII<sup>9</sup> (Fig. 4d), NAO<sup>10</sup> (Fig. 4e).

The above list of robots used in educational robotics and robot competitions shows that the platforms are often fixed, wheeled and equipped with common sensors such as light, distance, touch etc. (Fig. 4a,b,c,d,f,g,h). The sensors and motors are usually connected to a central control unit. The user programs the controller of the robot on a PC using C/C++, Java or derived simplified programming languages and uploads the code to the robot. Besides Botball (Fig. 4f) which is a robot kit composed out of several different platforms, LEGO NXT

(Fig. 4i) is the most open and flexible one regarding the shape of the robots that can be built.

#### B. Mechanical Construction Kits

We use the term “mechanical construction kit” synonymously with “model construction kit”. With these terms we refer to construction systems usually comprising re-usable elements such as strips, plates, angle girders, axles and gears with nuts and bolts to connect the pieces. The elements can be made out of plastic or metal, the connections can be screwed or stuck. We distinguish between “construction sets” and “construction kits”. A construction set has a determined and fixed set of elements which can be assembled into one specific object (e.g. a truck) by following an assembly guide. On the contrary, a construction kit has a variety of different elements to enable the construction of any object possible within the constraints of the elements at hand and the imagination of the user.

The history of classical mechanical construction kits goes back to the beginning of the last century with Frank Hornby who invented and patented 1901 a new toy called “Mechanics Made Easy”, also known as “Meccano”. Since then a variety of similar products emerged such as Eitech, Märklin or Stokys, some compatible with the 0.5 inch (1.273 cm) spacing of Meccano. Basically all of the traditional manufacturers suffered lately from decreasing interest in these kind of toys and the takeover of other construction kits such as LEGO. From the traditional manufacturers that survived until today, many still do not support interfaces to standard actuators and sensors (some provide a limited selection of proprietary motors).

Table I lists a collection of mechanical construction kits

<sup>5</sup><http://mindstorms.lego.com>

<sup>6</sup>[http://www.arexx.com/arexx.php?cmd=goto&cparam=p\\_asuro](http://www.arexx.com/arexx.php?cmd=goto&cparam=p_asuro)

<sup>7</sup><http://www.arduino.cc>

<sup>8</sup><http://www.e-puck.org>

<sup>9</sup><https://aseba.wikidot.com/en:thymio>

<sup>10</sup><http://www.aldebaran-robotics.com>

from manufacturers from all over the world. A short description about each construction kit is given in the respective table column. Further, we list some advantages and disadvantages of each platform that will be taken into consideration for the later sections of this paper. Even though some of the platforms provide robot controllers, we focus here on the pure mechanical properties of a kit.

We took the following for us important criteria into consideration while evaluating the mechanical construction kits:

- **flexibility:** the construction of different robots should be possible, it should therefore not be a construction set.
- **compatibility:** the kit should preferably have industrial standards e.g. metric threads.
- **stability:** the assembled system should be stable, therefore preferably screwed connections. The parts should not wear out easily.
- **low-cost:** the product should have a reasonable price.
- **availability:** each single component of a kit should be individually available for purchase.

The list of mechanical construction kits in Table I shows that there is not a great variety of construction concepts. Eitech, Meccano and Stokys for instance are very similar (Eitech is compatible to Meccano's hole size whereas Stokys took over Meccano's hole spacing). This might be explained by the common period when the companies were founded. Also Mindstorms NXT and ROBO are very similar: the robot is basically built around the main controller unit. Lynxmotion differs the most, since it focuses mainly on building joints. This on the other hand constraints the possible shapes that can be realized. It can be said that generally an object is built around an initial base plate of different sizes (or around a controller unit). The components are stuck or screwed on that base plate. Basically all construction kits (except Makeblock) force the user to connect the attached components according to their fixed hole spacing grid, additionally they often provide very few varieties of angles (often 90° or 45°). None of the listed mechanical construction kits provide passive dynamic elements (except the spring in Lynxmotion) and unconventional, soft materials.

Based on the advantages and disadvantages shown in Table I, we decided to pick Stokys and LEGO as the two base construction kits for the extensions described in the following section. We took LEGO because it is widely used in educational robotics and Stokys because each part can be purchased individually and it uses metric hole sizes (this is more convenient when located in Europe and it's compatible with the LocoKit rod size). Nevertheless, the examples we show in the following sections can also be transferred to some of the other listed construction kits.

### III. INTERFACE SOLUTIONS BETWEEN MECHANICAL CONSTRUCTION KITS

There is no universal robot kit or mechanical construction kit that meets each user's particular need. Therefore, it makes more sense to combine different products to achieve more flexibility. Sometimes the LEGO NXT robotic components

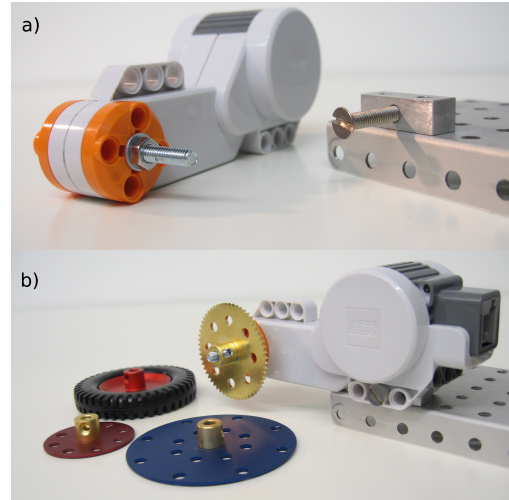


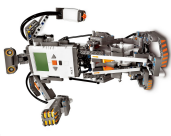







Fig. 5. a) A metric M4 screw is screwed in a LEGO motor in order to get a more stable motor shaft. A small customized block-shaped part designed to interface a LEGO motor (or sensor) with a mechanical construction kit (Stokys or Meccano). b) By means of the custom block-shaped part and the screw shaft a variety of Stokys elements can now be attached to the LEGO motor. Stokys uses standardized, metric hole sizes which increase flexibility to add other off-the-shelf components.

such as the controller and the motors are fine but the mechanical construction has to be more stable than plastic parts stuck together. Interfacing mechanical construction kits such as Meccano or Stokys could be a solution. On the other hand these traditional construction kits do not provide any interfaces to standard actuators in case the user would like to use DC motors or servo motors. The following sections show some simple solutions to these problems.

#### A. Solutions to Interface LEGO with Stokys

Everyone is familiar with LEGO blocks. The LEGO NXT kit provides a variety of bricks, connectors, wheels, rubber parts etc. Assemblies can be built and changed quickly and easily. However, since these parts are not screwed and the material is plastic they might not be precise enough or wear out too quickly for some applications. Fig. 5a presents an example where a metric M4 screw is screwed in a LEGO motor in order to get a more stable motor shaft. Despite the screw thread that is created within the motor due to this procedure, the original LEGO shaft can still be used (the star-shape hole remains). We had to produce a small customized part (little block mounted on a base part on the right side of Fig. 5a) in order to interface the LEGO motor with a mechanical construction kit (Stokys). By means of this custom metal part and the screw shaft a variety of Stokys components can now be attached to the LEGO motors, sensors and other LEGO components (Fig. 5b). Stokys uses standardized, metric hole dimensions which increase flexibility to add other off-the-shelf components.

TABLE I  
A SELECTION OF MECHANICAL CONSTRUCTION KITS

Product	Picture	URL	Type	Advantages	Disadvantages
Lego (DK) <b>Mindstorms NXT</b>		<a href="http://mindstorms.lego.com">mindstorms.lego.com</a>	Comprehensive mechanical construction kit made out of plastic. Includes actuators, sensors, a control unit. Provides graphical and classical programming. Stuck connectors.	Great flexibility, robustness of hardware and software, large variety of sensors, availability of individual parts, big community, variety of educational materials, general familiarity with LEGO blocks, 3rd-party additions, graphical programming and classical programming	Plastic parts wear out, stuck connections aren't stable enough for high impact applications (e.g. legged locomotion), proprietary mechanical interfaces (e.g. plugs, hole size), only 3 motor outputs and 4 sensor inputs per controller, big size and heavy weight (esp. controller).
Pitsco Education (USA) <b>Tetrix</b>		<a href="http://www.tetrixrobotics.com">www.tetrixrobotics.com</a>	Aluminum mechanical construction kit that includes structural material, actuators, gears, motor controllers and wheels. It provides interface solutions with LEGO Mindstorms NXT controller as well as LEGO blocks. Screwed connectors.	Great flexibility, availability of individual parts, robustness due to screwed connectors and aluminum material.	Imperial system (not metric), rather expensive, no soft or passive components.
Fischertechnik (GER) <b>ROBO</b>		<a href="http://www.fischertechnik.de">www.fischertechnik.de</a>	Plastic construction kit with actuators, sensors, and a controller. Graphical and classical programming. Sticked connectors.	Flexibility, easy assembly, a variety of sensors and unusual actuators, such as pneumatic motors	Plastic parts wear out and might not be stable enough for high impact applications, limited variety of construction parts, construction parts aren't usually sold individually, proprietary mechanical interfaces.
Lynxmotion (USA) <b>Lynxmotion</b>		<a href="http://www.lynxmotion.com">www.lynxmotion.com</a>	Mechanical construction kit made out of aluminum. Provides additionally actuators, sensors, grippers, springs, controllers, and a GUI. Screwed connectors.	Good quality material, stability, availability of individual parts, use of standard motors.	Imperial (not metric) system, rather expensive, very limited variety of components, no soft components.
Makeblock (CHN) <b>Makeblock</b>		<a href="http://makeblock.cc">makeblock.cc</a>	New mechanical construction kit made out of aluminum. Compatible with industrial standards. Screwed connectors.	Stability, originality due to flexible screw positioning and adjustable components, provides interfaces to standard actuators, interfaces with Arduino, relatively cheap price.	No passive or soft components, product is still new and under development.
Eitech (GER) <b>Eitech construction</b>		<a href="http://www.eitech.de">www.eitech.de</a>	Mechanical construction kit made out of steel. Provides additionally DC motors, gears, wheels, solar panels. Screwed connectors. Metric hole spacing (1 cm).	Stability, variety of elements, unconventional power supply (solar panel), cheap price, compatible hole size with Meccano and Stokys.	Relatively heavy weight (steel), no passive or soft components, no standard actuator interfaces (only to proprietary motors), hole spacing incompatible with Meccano or Stokys.
Meccano (UK) <b>Meccano</b>		<a href="http://www.meccano.com">www.meccano.com</a>	Traditional construction kit made out of steel. Provides DC motors, gears and wheels. Imperial system hole spacing (1.273 cm) and hole size (4.2 mm). Screwed connectors.	Stability, variety of elements, cheap price, generally known in Europe.	Relatively heavy weight (steel), no passive or soft components, no interfaces to standard actuators, components usually not sold individually.
Stokys (CH) <b>Stokys</b>		<a href="http://www.stokys.ch">www.stokys.ch</a>	Mechanical construction kit made out of aluminum. Provides additionally gears, wheels, DC motors. Imperial system hole spacing (1.273 cm) but metric hole size (4 mm). Screwed connectors.	Stability, availability of individual parts, compatibility with Meccano hole spacing and hole size (only 0.2 mm difference), variety of elements, light weight (aluminum).	No passive or soft components, no interface to standard actuators, relatively expensive (aluminum has its price).



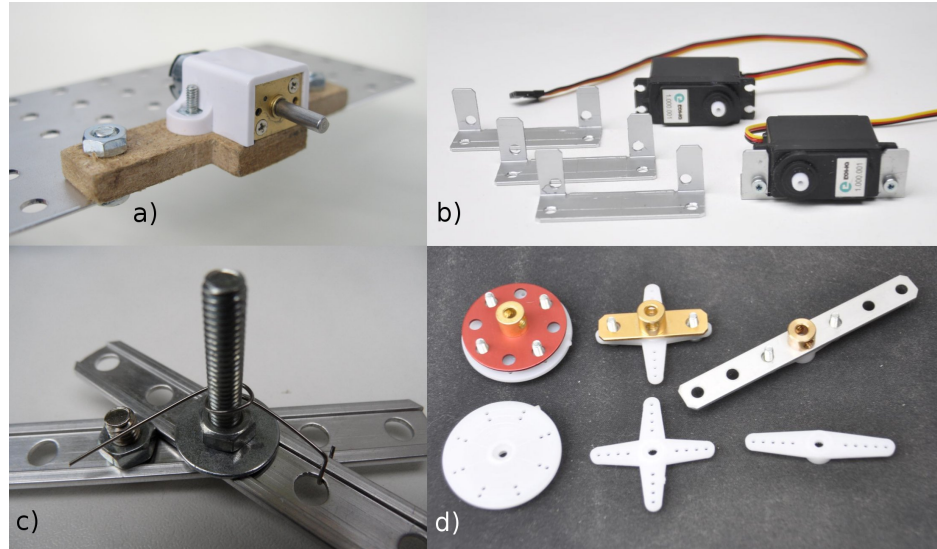


Fig. 6. A collection of different ways of creating interfaces between mechanical construction kits. a) Proprietary DC motor casing screwed onto a Stokys element by means of a custom made wooden connector part. b) A custom aluminum part designed for a servo motor to be screwed onto a Stokys or Meccano plate. c) A passive dynamic joint created with a piece of spring steel. d) Proprietary servo wheels screwed onto Stokys elements to enable a greater variety of possible servo wheel extensions.

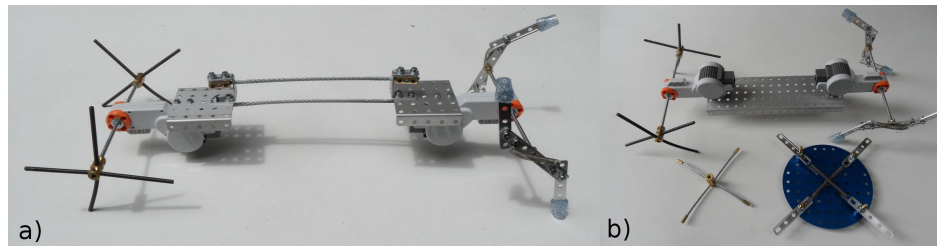


Fig. 7. Classical mechanical construction components have been extended with cheap, widely available material to create passive and flexible properties. a) A flexible spine can be created by using a piece of steel rope. b) A set of unconventional wheels made out of stings from speedometers, plastic tubes and steel ropes.

#### B. Solutions to Interface Stokys with Actuators

The casing of servo motors are not standardized and vary with each motor type. This is problematic since servo motors break easily when frequently used in a class environment. If the very same servo type is not available, often the replacement servo does not fit into the current setup. Fig. 6b shows a custom aluminum part designed for a servo motor to be screwed onto a Stokys or Meccano plate. Fig. 6a shows a proprietary DC motor casing screwed onto a Stokys element by means of a custom made wooden connector part. Fig. 6d shows proprietary servo wheels screwed onto Stokys elements to achieve a greater variety of possible servo wheel extensions.

The CAD files of all custom parts described in this paper

are available on our website<sup>11</sup>.

#### IV. SOLUTIONS TO EXTEND STOKYS WITH PASSIVE DYNAMIC MATERIALS

To build robots as in Fig. 1, 2 we need to incorporate passive dynamic materials. Structural elements that possess these kind of characteristics are usually not supported by robot kits or mechanical construction kits. Our goal was to achieve this with easy available materials, so we tried plastic tubes, spring steel, steel ropes, and strings from speedometers usually used in motorcycles. Fig. 6c demonstrates how a simple piece of spring steel wired around a joint can achieve passive dynamic properties. The stiffness can be varied easily.

<sup>11</sup><http://www.embed-it.ch>

The advantage of steel rope is that it is stable, flexible but not fully elastic. It suits perfectly to construct a flexible spine for walking robots, see Fig. 7a. Figure 7b shows a selection of unconventional wheels made out of strings from speedometers, plastic tubes and steel ropes. LEGO actuators are interfaced with Stokys or Meccano according to the interface description in Fig. 5.

Fig. 8 shows a Theo Jansen Strandbeest robot built with the Stokys construction kit. Even though it is a lot bigger and heavier than the 3D printed version in Fig. 3, it is nice to see that it is possible at all to build a robot like that using solely one mechanical construction kit.

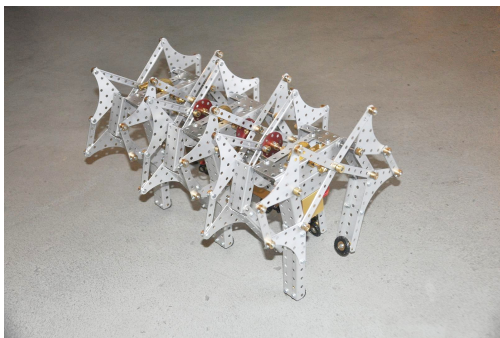


Fig. 8. An actuated version of Theo Jansen's Strandbeest built with Stokys construction kit. Image courtesy of Stokys Systeme AG.

## V. NEW TRENDS WITH THE LOCOKIT ROBOTIC CONSTRUCTION KIT

The philosophy behind this system is embodiment and that the interplay between individual components of the system have to work together to form dynamic locomotion. Being a construction kit, it enables the user to make adjustments to the robot after it has been built. Opposed to other systems, LocoKit does not constrain the user to place components at fixed positions or to use determined sizes of structural elements. In the review section of this paper we saw that mechanical construction kits usually have a fixed grid size of 1 cm or more. LocoKit enables the user to adjust the position of a component within a range of a few millimeters. Hereby, the user can explore how changes of the morphology effects the performance of the system on a very fine scale. Examples of such changes could be body width, leg length, center of mass, angle of attack etc.

LocoKit distinguishes itself from the other construction kits mentioned earlier in this paper, by being the only one directly targeted to walking, running or jumping robots (Fig. 10). Also, by being designed with a focus on non-rigid elements, it gives the user the opportunity to build robots, where the body is not rigid but bendable. This feature is controversial because rigid systems are often preferred since they are easier to model and control. However, the aim of LocoKit is to be a system that

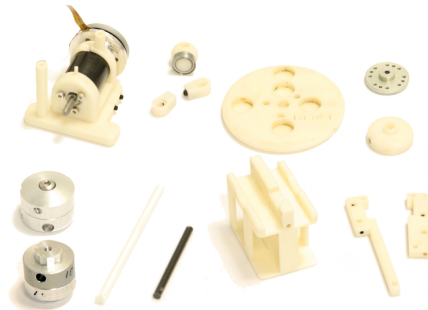


Fig. 9. A selection of the most important mechanical LocoKit parts.

supports the creating of model-free, bottom up robots with limited need for a mathematical model to describe the system beforehand.

Everything in the LocoKit system is designed such that it fits to a 4 mm rod (Fig. 9). For now, these rods are mainly composed of fiberglass or carbon fiber but could in theory be made of any material as long as it forms a 4 mm round rod. The reason for this design choice is that the user is more free to choose other materials, e.g. more soft, rigid, lighter or heavier ones. It also opens up the opportunity of making some parts of the structure stiff and other ones soft, depending on the kind of desired structure.

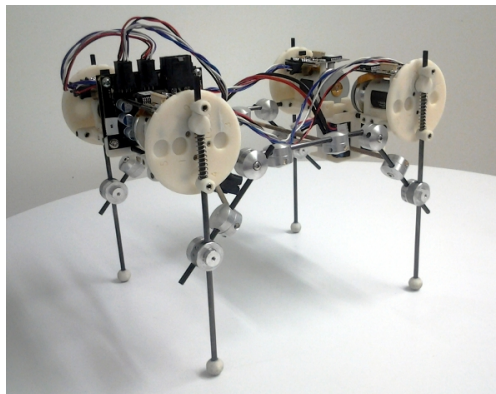


Fig. 10. A quadruped robot built with the LocoKit construction kit of Fig. 9. The used structural materials are all bendable, enabling the body to bend under its own weight. A more slip-like walking pattern is achieved due to the springs located in the upper part of the legs.

This system is still under development and therefore not yet commercially available. For more in dept information, see Larsen et al. [15].

## VI. CONCLUSION

In this paper we described our need for unconventional robot morphologies for teaching embodied artificial intelligence and biologically inspired robotics. We described the robots we usually use in class which are custom built by means of expensive procedures such as 3D printing. Our aim is to replace the 3D printed parts using cheap, commercially available materials. After a short review on educational robot kits and mechanical construction kits we selected Stokys and LEGO as example kits. We described how common actuators can be interfaced with those proprietary toolkits by using easy custom made components. Further, we described examples on how to incorporate passive dynamic properties and compliant materials to those systems. In addition, we introduced the LocoKit, a new toolkit which is currently under development and which aims at providing those required properties for walking robots. Generally, we are pleased about the number of the toolkits available. However, we hope that in the future the manufacturers will go more towards open, standardized interfaces rather than proprietary hardware and software.

## VII. ACKNOWLEDGEMENTS

This project was supported by the European FP7 project LOCOMORPH under the grant agreements No. 231688.

## REFERENCES

- [1] Pfeifer, R., Bongard, J. *How the Body Shapes the Way We Think - A new View of Intelligence* MIT Press (2007) ISBN 978-0-262-16239-5
- [2] Pfeifer, R., Lungarella, M., Iida, F. *Self-organization, embodiment, and biologically inspired robotics* Science 318, 1088-1093
- [3] Pfeifer, R., Gomez, G. *Morphological computation - connecting brain, body, and environment* In B. Sendhoff, O. Sporns, E. Krner, H. Ritter, & K. Doya, K. (eds.), *Creating Brain-like Intelligence: From Basic Principles to Complex Intelligent Systems* (pp.66-83). Berlin: Springer.
- [4] Iida, F., Gomez, G., Pfeifer, R. *Exploiting body dynamics for controlling a running quadruped robot* In proceedings of the 12th International Conference on Advanced Robotics, ICAR (2005)
- [5] Rinderknecht, M., Ruesch, J., Hadorn, M. *The Lagging Legs Exploiting Body Dynamics to Steer a Quadrupedal Agent* In proceedings of the International Conference on Morphological Computation (ICMC), Venice, Italy, March 2007
- [6] Saranli, U., Buehler, M., Koditschek, D. E., *RHex: A simple and highly mobile hexapod robot* International Journal of Robotics Research, vol. 20, no. 7, pp. 616-631, 2001.
- [7] Assaf, D., Pfeifer, R. *EmbedIT - An Open Robotic Kit for Education* Research and Education in Robotics - EUROBOT 2011, Communications in Computer and Information Science, Springer Volume 161, pp. 29-39, (2011)
- [8] Alimisis, D., Moro, M., Arlegui, J., Frangou, S., Papanikolaou, K. *Robotics & Constructivism in Education: the TERECoP project*, In Proceedings of EuroLogo 2007, Bratislava, (2007)
- [9] Bredenfeld, A., Leimbach, T. *The Roberta Initiative*. In Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010), pp. 558-567, ISBN 978-3-00-032863-3
- [10] Bredenfeld, A., Hofmann, A., Steinbauer, G. *Robotics in Education Initiatives in Europe Status, Shortcomings and Open Questions*. In Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010), pp. 568-574, ISBN 978-3-00-032863-3
- [11] Kabatova, M., Pekarova, J. *Lessons learnt with LEGO Mindstorms: from beginner to teaching robotics*. AT&P Journal Plus 2, Robotics in Education, pp. 51-56, ISSN 1336-5010, (2010)
- [12] Menegatti, E., Moro, M. *Educational Robotics from high-school to Master of Science*. In Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010), pp. 639-648, ISBN 978-3-00-032863-3
- [13] Balogh, R. *Acrob - an Educational Robotic Platform*. AT&P Journal Plus 2, Robotics in Education, pp. 6-9, ISSN 1336-5010, (2010)
- [14] Assaf, D., Pfeifer, R. *Robotics as Part of an Informatics Degree Program for Teachers*. In Proceedings of Society for Information Technology & Teacher Education International Conference 2011 (pp. 3128-3133). Chesapeake, VA: AACE. (2011)
- [15] J. Larsen, D. Brandt, K. Stoy, *LocoKit: A Construction Kit for Developing Robot Morphologies*, in Proceedings of the 12th International Conference on Adaptive Behaviour (SAB2012)





# Teaching Soft Robotics: A Case Study in Building Elastic Creatures that Locomote.

Submitted article:

Assaf, D. and Yu, X. and Wang, L. and Iida, F. (2012)

# Teaching Soft Robotics: A Case Study in Building Elastic Creatures that Locomote

Dorit Assaf, Xiaoxiang Yu, Liyu Wang, and Fumiya Iida

**Abstract**—Locomotion has been one of the frequently used case studies in hands-on curricula in robotics education. Students are usually asked to construct their own wheeled or legged robots from modular robot kits. In the development process of a robot students tend to emphasize on the programming part and consequently, neglect the design of the robot's body. However, the morphology of a robot (i.e. its body shape and material properties) plays an important role especially in dynamic tasks such as locomotion. In this paper we introduce a case study of a tutorial on soft robotics where students were encouraged to focus solely on the morphology of a robot to achieve stable and fast locomotion. The students should experience the influence material properties exert on the performance of a robot and consequently, extract design principles. This tutorial was held in the context of the 2012 Summer School on Soft Robotics at ETH Zurich, which was one of the world's first courses specialized in the emerging field. We describe the tutorial set-up, the hardware and software, the students' assessment criteria as well as the results. Based on the high creativity and diversity of the robots built by the students, we conclude that the concept of this tutorial has great potentials for both education and research.

**Index Terms**—soft robotics, robotics education, robot competition, robot locomotion, robotic kits

## I. INTRODUCTION

ROBOTS are a popular tool to teach science, technology and engineering to students at levels of education, which range from kindergarten [39], [30], [41] and primary schools [7], over secondary schools [12], [26], [25] to universities [21], [1], [19] [35], [14], [34]. There are a large number of off-the-shelf educational robot kits available such as LEGO NXT<sup>1</sup>, Bioloid<sup>2</sup>, Nao<sup>3</sup> or ThymioII<sup>4</sup>, each targeted to a specific audience or robotic application. Common assignments in robot classes and competitions are legged locomotion (using humanoids such as Nao), obstacle avoidance or navigation in space while picking and placing objects. This might be explained by the typical engineering nature of these tasks and their easy evaluation. Fixed robotic platforms (car like vehicles or humanoids) are being used when the students have to emphasize on the programming task of the assignment, whereas modular platforms such as LEGO NXT [17], [31] or Bioloid [6] are chosen when the focus lies on both design

of the morphology and development of the control. Even in the latter case students tend to invest more time in the programming task than in the design of the robot's body. The custom robots built by the students using modular robot kits are often biased by the classical fixed platforms, and therefore often look similar. The students are usually not aware of the important role the morphology plays in the performance of a robot and therefore rather try to implement a control solution for a problem. The notion of embodied intelligence states that behavior is not the result of control processes only, but of a subtle interplay between control, body (morphology and materials) and the environment [28], [27]. We use the term morphological computation to designate the fact that morphological and material characteristics of an organism can take over a large part of its functionality (e.g. the passive forward swing of the leg in walking, the spring-like properties of the muscles, the weight distribution etc.), and therefore reduce computational complexity [29]. Brains and bodies of organisms evolved together and function highly interdependent, a fact that should always be kept in mind while studying or developing intelligent systems. A comprehensive robotics education that fosters equally both building and programming skills should be targeted where possible [23], [5], [32], [13]. Further, conventional robotic platforms use, next to their emphasis on control engineering tasks, preferably classical, rigid materials such as aluminum, steel or plastic [4]. Biological systems, however, incorporate passive and compliant elements, material properties which have been proven to be very effective for artificial systems as well. As more and more robots will be used outside traditional industrial environments, safe interaction with vulnerable subjects such as humans is required. The emerging research field of soft robotics addresses these issues by the use of soft or compliant elements to overcome the limitations of traditional rigid systems. For more details about this research field, see section II-A.

In the context of the 2012 ETH Summer School on Soft Robotics we designed a novel hands-on tutorial where 59 international undergraduate and postgraduate students had the opportunity to build their own elastic creatures for locomotion. Our aim was to emphasize entirely on the morphology of the robot and therefore disabled any programming possibilities. Since the research field is relatively new, there are not yet well established design principles for soft-bodied robot locomotion that could be taught. Our aim was besides the learning experience for the students while building the robots, to take advantage of the produced output and utilize it for further studies. The idea is to consider this tutorial as an iterative

X. Yu, L. Wang and F. Iida are affiliated with Bio-Inspired Robotics Lab (BIRLab), Institute of Robotics and Intelligent Systems, ETH Zurich, Switzerland. e-mail: xiaoxiang.yu@mavt.ethz.ch, liyu.wang@mavt.ethz.ch, iidaf@ethz.ch

D. Assaf is affiliated with Artificial Intelligence Laboratory, Department of Informatics, University of Zurich, Switzerland. e-mail: assaf@ifi.uzh.ch

<sup>1</sup><http://mindstorms.lego.com>

<sup>2</sup>[http://www.robotis.com/xe/bioloid\\_en](http://www.robotis.com/xe/bioloid_en)

<sup>3</sup><http://www.aldebaran-robotics.com>

<sup>4</sup><https://aseba.wiki.net/en:thymio>

process whose creatures evolve towards more sophisticated body designs, as design principles are derived from the output of each instance and taught during the following.

In the related work section we introduce the research field of soft robotics as well as some educational principles we took into considerations while designing the tutorial. Section III describes the design of the tutorial, its used hardware and software. This is followed by the results (section V), discussion (section V), conclusion and future work (section VI).

## II. RELATED WORK

### A. Soft Robotics

We speak of soft-material robots when their physical system is entirely or partly made of intrinsically soft material. They build a subcategory of the so called soft-robots. Early days research in soft-material robot locomotion focused on the development and application of soft actuators [40]. More recently, interest aroused in soft-bodied locomotion, where the actual body of the robot consists of soft materials (despite the actuators that may be shape memory alloys or compressed fluids) [24], [18], [38], [42], [36], [11], [20], [10], [37]. By exploiting the (visco)elasticity of the soft material in interaction with the environment, robots have demonstrated various forms of terrestrial locomotion such as legged walking, limbless crawling as well as rolling. Since soft-bodied robots are claimed to be compliant and adaptive towards different shapes of environments, their locomotion behaviors are difficult to be modeled due to the complexity of modeling continuum mechanics.

### B. Pedagogical Principles

Discovery learning [9] is a method of inquiry-based instruction which describes a constructivist learning approach where students are presented with a problem or question. The students are required to work independently to discover knowledge by themselves. The teacher is a facilitator which helps students along this process rather than providing direct instructions. In discovery learning situations students draw their own conclusions by exploring and manipulating objects, addressing questions, and performing experiments. Although the popularity of this form of instruction, its efficacy is controversial [22]. We adopted this learning method for our tutorial mainly because direct knowledge transfer from the teacher was yet not possible. Soft-bodied robot locomotion is part of an emerging research field, where studies on e.g. design principles and system dynamics are still on-going. Through discovery learning not only the students may gain a deeper understanding, but also the teachers could draw valuable conclusions for their research purposes.

## III. DESIGN OF THE TUTORIAL

### A. The concept of the exercise

Soft robotics is a highly interdisciplinary field which encompasses not only robotics but also disciplines such as physics, applied material science or biology. Therefore, we

targeted an inhomogeneous crowd of students, basically anybody interested in learning about soft-bodied robot locomotion was welcome. However, due to the diverse backgrounds of the students, identical prerequisites could not be assumed such that the tutorial had to be self-contained. Further, the large number of students and the limited class time was a critical constraining factor in the design process of the tutorial. Usually, hands-on exercises are besides the high material costs very time consuming in preparation. We decided to keep the exercise simple, cheap in price at the same time encouraging, pedagogically valuable and fun.

Our laboratories have a tradition in teaching embodied intelligence to students with different backgrounds. In several other teaching activities we conducted an exercise using LEGO NXT robots, from which we usually received positive feedback [3]. It was designed as a competition where each team received a default quadrupedal LEGO robot (the initial shape of the robot is inspired by [33]). The control program of the robot was fixed and therefore could not be altered (both motors turned forwards at a constant speed). The teams were asked to achieve faster locomotion by solely changing the robot's morphology, with the constraint, that no additional LEGO construction parts were allowed to be attached (construction parts were allowed to be left out though). The aim was to encourage students to emphasize exclusively on the shape of the robot and its dynamics while interacting with the environment. This exercise proved to be very helpful in the dissemination process of the embodiment paradigm. Based on the positive experiences with this exercise we decided to take over its concept for our tutorial.

### B. Choosing a robotic platform

LEGO NXT is an excellent platform for schools, nevertheless, we noticed several drawbacks while using it for our purpose. After each exercise we rebuilt the initial configuration of the LEGO robot to be ready for the next instance of this exercise. The same construction parts were reused many times and consequently, the connectors wore out. The robots became unstable and frequently fell apart while running. The rigid, plastic material isn't suitable for high impact forces acting on the robot, which is especially the case during legged locomotion. Another constraint is the connector design of the LEGO construction parts. LEGO uses proprietary hole size and hole spacing dimensions. The hole sizes hinder the attachment of other cheap, standardized, off-the-shelf parts such as metric screws, axles or girders. Vernier adjustment is not possible due to the fixed hole spacing as well as the specific locations of attachment points on both controller and motor. Angles, lengths and widths of elements cannot be altered, least of all the availability of compliant and soft elements. All these constraints are critical for us, since we aim at varying the shapes of the robots as much as possible. Due to the rather expensive price of the LEGO NXT controller and motors, we usually could only conduct this exercise with a small number of students. Based on these drawbacks we decided to look for another platform to be used in this tutorial. Since none of the commercially available robot kits and mechanical

construction kits provide soft and compliant materials [4], we chose a familiar and cheap solution that were developed in our laboratories: The BIRLab's research projects frequently use adhesives for their robots (e.g. the gripper) [8], while the AILAB is currently developing an open-source robotic kit for education "EmbedIT" [2]. So we combined the adhesives together with some modules of the EmbedIT toolkit to create the desired soft-robotic platform to be used in class. Further details about the hardware and software are given in the following section.

### C. Hardware and software

The EmbedIT robot kit consists of three different modules (master, actuator, sensor) which can be added and removed from a CAN bus. The modules are small printed circuit boards (compatible with Arduino<sup>5</sup>), each of which is designed for its individual purpose (actuation, sensing). The user can add any number of actuator and sensor modules to the bus and additionally, access and control each of them through the respective graphical user interface (GUI), the master module and a wireless communication connection. For this tutorial only two servo motors were needed, so only one actuator module was sufficient (it can control up to four servo motors). The GUI enables the user to save continuous or discrete servo positions directly to the module's nonvolatile memory (EEPROM) without programming (Figure 2). A sinusoidal function (oscillation between 0° and 180°) is preprogrammed in the actuator module and through the GUI the user can change the sine parameters (amplitude, frequency, offset, phase Lag) remotely and in real time. Once the desired sine parameters are found and saved to the EEPROM, the master module can then be removed from the bus to leave the actuator module run autonomously. This user friendly way to access servo motors remotely and save simple, ready-made control procedures directly, reduces the preparation overhead of the tutorial drastically. It is actually done in just a few minutes.

The starting position for each team were two servo motors oscillating between 0° and 180° at approximately 1 Hz. We decided to use servo motors instead of fully rotating DC motors in order to reduce the temptation to construct wheels. The actuator board supports up to four servo motors, so the student could chose between two oscillation modes (synchronous and asynchronous, i.e. shifted phase lag) by plugging the motors into their respective sockets.

The main construction elements to build the robot's soft body were thermoplastic adhesives (i.e. hot glue sticks), normally used in glue guns. The students were allowed to cut or bend the sticks arbitrarily, in order to use them as elastic support elements of the robot. At the same time the sticks could just be melted into adhesives to attach them together or to the few rigid parts of the robot (7.4V Li-Po battery, PCB, RS-2 MODEL CRAFT servo motors). Generally, the students were allowed to use any number of glue sticks and as many of the other accessory tools which were cable straps and duct tape. Figure 1 shows a snapshot of all components initially provided to one team. With a competitive price of

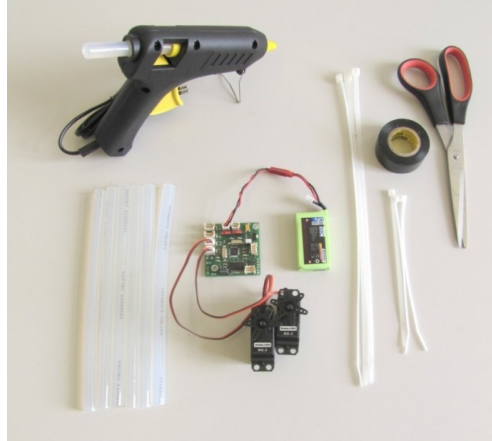


Fig. 1. All components initially provided to one team: A glue gun, glue sticks, EmbedIT actuator board with LiPo battery and two servo motors attached, long and short cable straps, insulation tape and a scissor.

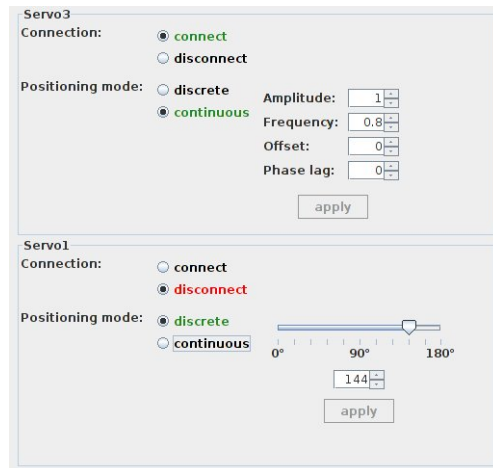


Fig. 2. The graphical user interface of the EmbedIT toolkit. With the servo motor control interface discrete and continuous servo positions can be set and applied directly to the hardware.

approximately 88 Swiss Francs (see Table I) for each robot our goal for a cheap solution was met. Besides, most of the expensive components can be reused, such that a more economic average price can be obtained if repeating the class multiple times. This low cost is mainly achieved by the simple electronics and materials as well as the exclusion of sensors, which also complies with the "cheap design" philosophy [27].

### D. Tutorial curriculum

The tutorial had to be self-contained at an introductory level and conducted within three hours. Initially, we gave a short overview about the EmbedIT toolkit and an introduction on

<sup>5</sup><http://www.arduino.cc/>

TABLE I  
A LIST OF COMPONENTS INITIALLY PROVIDED TO ONE TEAM TO BUILD A  
SOFT-BODIED ROBOT

Component	Quantity	Price per unit (USD)
EmbedIT actuator board	1	33
LiPo Battery 7.4V	1	10.5
Servo motors	2	5.2
Glue stick	Any number	5.5 (averagely used)
Glue gun	1	15.8
Insulation tape	1	6.3
Scissor	1	3.9
Cable strap long	Any number	1 (averagely used)
Cable strap short	Any number	2 (averagely used)
<b>Total</b>		<b>88.4</b>

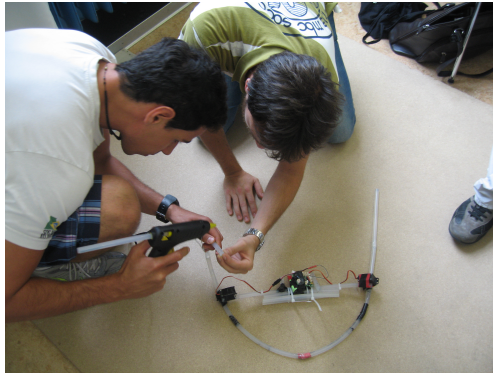


Fig. 3. Students working on a soft-bodied robot.

soft-robots. We explained the goal of the exercise and the competition rules: A 1.2 meter long (and 1 meter wide) lane was marked on a wooden board. Every team had three official attempts where the time the robot spent to travel those 1.2 meter was measured and recorded. If the robot left the official path sideways, the attempt was not valid and therefore not successful. The fastest result was saved and counted for the competition and the winner group received a prize. If time was left the groups were allowed to disassemble their robot after the three attempts in order to build a new one (Figure 3). Figure 4 shows a scene of the class.

#### IV. RESULTS

In the summer school, 59 international students from different disciplines and academic levels participated in the tutorial. We split the students (alphabetically by surnames) in two classes of 30 and 29 students respectively. The same three hour tutorial was held for both groups on different days. The students formed teams consisting of two or three people, such that each tutorial had ten teams. The students were asked to fill out a questionnaire with questions about their backgrounds and an evaluation survey. Figure 5 summarizes the statistics of academic levels and backgrounds of the attendees. Figure 5a shows, that doctoral students occupy the largest part of the participants, followed by undergraduates, other research staff,



Fig. 4. The class scene of the tutorial on soft-bodied robot locomotion in the context of 2012 ETH Summer School on Soft Robotics.

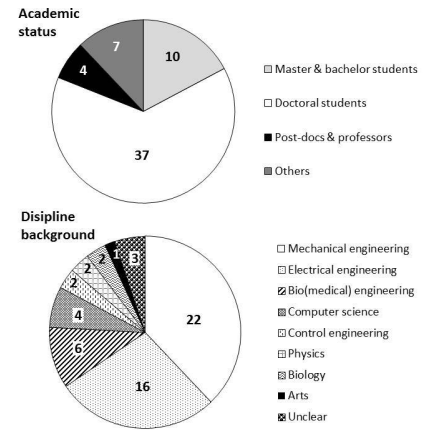


Fig. 5. Academic levels and backgrounds of the students participated in the tutorial on soft-bodied robot locomotion.

postgraduates and professors. Electrical engineering and mechanical engineering were the mostly mentioned backgrounds besides other engineering related disciplines such as control engineering or computer science. However, students attended also from a variety of non-engineering fields (e.g. arts, physics or biology).

##### A. Performance of the robots

Table II presents statistics of each tutorial class in its respective table column. Both separately held tutorials gave approximately equivalent results, which implies that the data is reasonable and the results can be accepted for analysis. The 10 teams in each tutorial class built 11 and 9 robots respectively. In class II one group did not finish on time whereas in class I one group had spare time to disassemble their first robot in order to build another one. Each team had three attempts to officially record and measure their robot's

TABLE II  
STATISTICS OF BOTH TUTORIAL CLASSES

	Class I	Class II
Number of students	30	29
Number of teams	10	10
Number of robots built	11	9
Number of attempts	21	20
Number of successful robots	6	6
Time spent for the fastest robot	3.6 seconds	4.5 seconds
Time passed for first attempt	70 minutes	75 minutes

TABLE III  
A SELECTION OF ROBOTS BUILT BY THE STUDENTS OF BOTH CLASSES  
AND THEIR PROPERTIES

Robot	length x width (cm x cm)	time spent (seconds)	locomotion	propel pattern
Figure 6a	21 x 31	3.6	rolling	side legs onto ground
Figure 6b	28 x 18	4.5	hybrid	rear elastic legs
Figure 6c	24 x 22	5.9	crawling	rear elastic legs
Figure 6d	21 x 22	9.8	crawling	rear elastic legs
Figure 6e	49 x 49	12.3	crawling	whole body with ground
Figure 6f	58 x 15	14	legged	side legs onto ground
Figure 6g	23 x 22	23.1	legged	side legs onto ground
Figure 6h	44 x 14	30.9	undulation	whole body with ground

performance. Each team did that twice within the duration of the tutorial, the first successful attempts occurred in both classes after approximately 70-75 minutes. In both classes 60% of the robots (class I 54.5%, class II 66.6%) successfully moved straight forwards to reach the 1.2 meters goal, all the other attempts were not valid and therefore not recorded. The main reasons for invalid attempts were cases when the robot did not move forwards at all or when it left the competition area sideways. Moving straight was the biggest difficulty due to imprecise body properties of the robot caused by the soft material and its assembly method. The fastest robots in both classes managed to travel the 1.2 meters in approximately 3.5-4.5 seconds.

#### B. Robot morphologies

Figure 6 shows a representative selection of robots constructed by the students in both tutorial classes. The morphologies can be classified in three major types: wheels (Figure 6a, b), bars with front mass (Figure 6c, d, f, g), and limbless (Figure 6e, h). The robot's locomotion types and propel methods are summarized in Table III. In both classes the wheeled robots were the fastest ones, however, the biologically inspired movements (e.g. Figure 6e, h) were the more interesting ones.

#### C. Students feedback

The students filled out questionnaires at the end of each tutorial session. In total, 58 questionnaires had been returned. Since this was the first time we held this kind of tutorial, we planned to obtain general feedback about the tutorial itself and asked for improvement suggestions. Figure 7 shows general feedback about the tutorial. We asked questions such as "did you like the tutorial?", "did the exercise help to understand the importance of morphology?", or "would you recommend this tutorial for other people?". The students could choose between a grade of 1 to 6, where 1 means "not at all" and 6 stands for "very much". Overall, the majority chose 5 or 6 as a grade for all questions, which is a satisfying outcome for this tutorial. However, for future tutorials we should improve the understanding of the importance of morphology.

Further, we asked for their self-evaluation about the difficulty of the task and the performance of their robot. Figure 8 shows the responses to the questions "was it difficult to build a robot with the provided components?", "was it difficult to achieve locomotion with the provided components?", and "did your robot perform well?". Again, the students could grade between 1 and 6, where 1 is "not at all" and 6 stands for "very much". Generally, students found the actual building of the robot with the provided components less difficult than the achievement of locomotion. This is not surprising since the building process using the glue gun is very straightforward. Ca. 58% of the students graded the performance of their robot quite good (between 4 and 6) which corresponds to the observed assessment results of the previous section IV-A, where 60% of the robots made it to the goal. Further, we asked whether the tutorial should be longer, same, or shorter in duration. 74% of the students approved that the duration of the tutorial should stay the same (3 hours). An equal number of students wished either longer or shorter durations.

Additionally, we gave the students the opportunity to make any suggestions about future tutorials in a free comment section of the questionnaire. The most mentioned suggestions were the wish to incorporate other criteria in the evaluation of the robot next to solely locomotion speed. Criteria mentioned were creativity of the design and bio-inspiration of the robot. Others would have liked to have more freedom in controlling the servos (more possibilities to change oscillation parameters). Further, it was often mentioned that design principles for soft-robots should have been taught in advance for direct application in the competition.

#### V. DISCUSSION

In the previous section we saw that a large number of successfully performing robots have been built from scratch within a short time. Therefore we consider the duration of tutorial was chosen appropriately. Further, a significantly larger variety of robot morphologies and movements have been developed by the students compared to the outcome the LEGO NXT exercise (section III-A) usually generates. This might be explained by the less restrictive platform chosen this time as well as by the soft material properties. Based on this first instance of the tutorial we learned that the wheeled robot shape

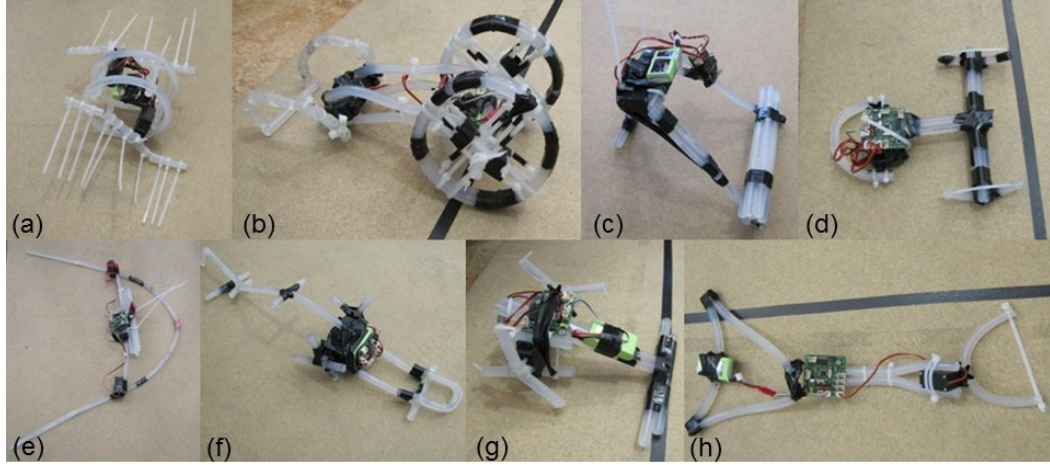


Fig. 6. A selection of robots built by the students within the three hour tutorial session. The morphologies can be classified in three major types: wheels (a, b), bars with front mass (c, d, f, g), limbless (e, h).

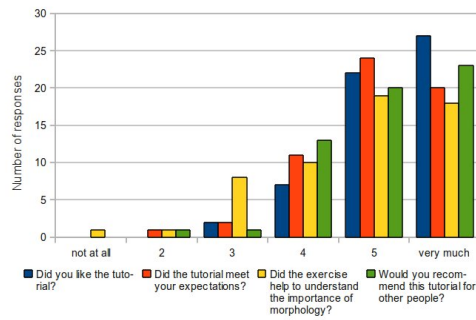


Fig. 7. A general feedback about the tutorial. Overall, a positive outcome since most of the students graded the questions with 5 or 6 (6 stands for “very much”). Improvements in the understanding of the importance of morphology should be aimed for future tutorials.

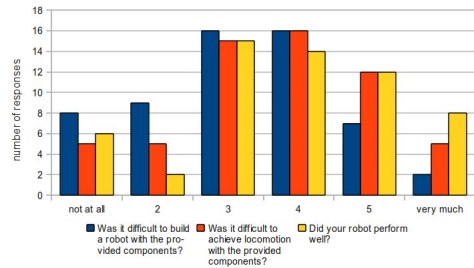


Fig. 8. Some questions about the self-evaluation of the difficulty of the task and the performance of the robot. The grade 1 is “not at all” and 6 stands for “very much”. Students found the building process of the robot easier than the achievement of locomotion. Ca. 58% of the students graded the performance of their robot between 4 and 6, which corresponds to the observed assessment results.

is the most efficient one regarding speed, even by using soft materials and oscillating actuators. This is interesting, since wheeled robots have been the fastest ones too in the LEGO NXT exercises, but there fully rotating DC motors had been used as well as rigid materials. The most significant challenge was to construct the robots such, that they follow a straight trajectory. This is clearly a drawback of the soft material.

As previously mentioned, the field of soft-bodied robot locomotion is still emerging. Therefore, neither well elaborated design principles exist nor best practices in body design for this task are known. It was consequently the right choice to design the tutorial according to the principle of discovery learning (section II-B). Based on the output generated by the students both teachers and students could discuss and extract design principles, as done in section IV-B of this paper. These design principles can then be presented as theory in future

iterations of this tutorial, which consequently could lead to improved designs, which at the same time acquires more data for further quantitative analysis. The cheap robotic platform as well as the short preparation time enables the teacher to conduct this tutorial for a large number of students. The more students the better the data, which is beneficial for this kind of “evolutionary” tutorial concept.

Furthermore, the students feedback was very positive. As wished by the students (and already as planned by ourselves), we will have to give an introduction into theoretical principles of soft-bodied robot locomotion for future instances of this tutorial. Further, we will have to focus more on the properties of the material at hand and on the importance of morphology, since the understanding of the importance of morphology was not as high as aimed. The students had mostly classical engineering backgrounds (mechanical engineering, electrical



engineering), thus the principles of embodied intelligence are not well known. Additionally, we might have to consider to value design creativity in the competition next to locomotion speed. This has been wished especially by those students that created interesting creatures which moved nicely but weren't among the fastest ones. This criteria would foster interesting, biologically inspired designs rather than solely the construction of wheeled robots. Wheeled robots could also be avoided by a redefinition of the task environment (e.g. rough terrain).

## VI. CONCLUSION AND FUTURE WORK

In this paper we introduced a novel curriculum for teaching soft-bodied robot locomotion to students with different backgrounds. A three hour self-containing tutorial was held in the context of the 2012 ETH Summer School on Soft Robotics. Since there are no well established design principles in building soft-bodied robots for locomotion, the aim was on the one hand to give a hands-on introduction in this emerging field, on the other to use the output produced by the students for analysis and derivation of design principles. We described the low-cost robotic toolkit we chose, which additionally significantly reduced the preparation effort for the teacher. Overall, we can be satisfied with the outcome of this first instance of the tutorial. Preliminary design principles could be derived which will be introduced in future iterations of this "evolutionary" tutorial.

Next tutorials are already planned from which we expect additional valuable data. The tutorial will be included in a robotics lecture at ETH institute of robotics and intelligent systems for undergraduates in fall term 2012, with an additional task to model the built robot. Further, numerous international teams will compete in building elastic creatures that locomote in the context of the Shanghai lectures<sup>6</sup>, which is a global lecture about natural and artificial intelligence held via video conference at about 15 Universities. The cheap price and the easy setup of the tutorial enables teams from all participating universities to join the competition.

## REFERENCES

- [1] Apiola, M., Lattu, M., Pasanen, T. A., *Creativity and intrinsic motivation in computer science education: experimenting with robots*, Proceedings of the fifteenth annual conference on Innovation and technology in computer science education, ITiCSE '10, 2010, pp 199–203, ACM, New York, NY, USA.
- [2] Assaf, D., Pfeifer, R. *EmbedIT – An Open Robotic Kit for Education Research and Education in Robotics – EUROBOT 2011*, Communications in Computer and Information Science, Springer Volume 161, pp. 29–39, (2011)
- [3] Assaf, D., Pfeifer, R. *Robotics as Part of an Informatics Degree Program for Teachers*. In Proceedings of Society for Information Technology & Teacher Education International Conference 2011 (pp. 3128–3133). Chesapeake, VA: AACE. (2011)
- [4] Assaf, D., Larsen J.C., Reichardt, M. *Extending Mechanical Construction Kits to Incorporate Passive and Compliant Elements for Educational Robotics*. In Proceedings of 3rd international conference on robotics in education (RIE 2012)
- [5] Baeroveldt, A.-J., Salomonsson, T., Astrand, B. *Vision-guided mobile robots for design competitions*, IEEE Robot. Autom. Mag., vol. 10, no. 2, pp. 38–44, Jun. 2003.
- [6] Bishop, B., Esposito, J., Piepmeier, J. *Moving without wheels: Educational experiments in robot design and locomotion* ASEE Computers in Education Journal, vol. 20, no. 3, pp. 41–49, Jul–Sep. 2010.
- [7] Bredendfeld, A., Leimbach, T. *The Roberta Initiative*. In Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010), pp. 558–567, ISBN 978-3-00-032863-3
- [8] Brodbeck, L., Liyu, W., Iida, F. *Robotic body extension based on hot melt adhesives* In Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 2012, pp. 43224327.
- [9] Bruner, J.S., *The act of discovery* Harvard Educational Review, vol. 31, no. 1, pp. 21–32, 1961.
- [10] Calisti, M., Giorrelli, M., Levy, G., Mazzolai, B., Hochner, B., Laschi, C., Dario, P. *An octopus-bioinspired solution to movement and manipulation for soft robots* Bioinspir. Biomim., vol. 6, no. 3, pp. 036002-1–11, Sep. 2011.
- [11] Cheng, N., Ishigami, G., Hawthorne, S., Chen, H., Hansen, M., Telleria, M., Playter, R., Iagnemma, K. *Design and analysis of a soft mobile robot composed of multiple thermally activated joints driven by a single actuator* in Proc. ICRA, 2010, pp. 5207–5212.
- [12] Church, W., Ford, T., Perova, N., Rogers, C. *Physics With Robotics: Using LEGO MINDSTORMS in High School Education* Educational Robotics and Beyond: Design and Evaluation pp 47–49, March 2010, AAAI Press, AAAI Spring Symposium Series.
- [13] Dias, M. B., Browning, B., Mills-Tetley, G. A., Amanquah, N., El-Moughny, N. *Undergraduate robotics education in technologically underserved communities* Proc. ICRA, 2007, pp. 1387–1392.
- [14] Dodds, Z., Greenwald, L., Howard, A., Tejada, S., Weinberg, J. *Components, curriculum, and community: Robots and robotics in undergraduate AI education* AI Mag., vol. 27, no. 1, pp. 11–22, Spring 2006.
- [15] Felder, R. M., Silverman, L.K., *Learning and teaching styles in engineering education* J. Eng. Educ., vol. 78, no. 7, pp. 674–681, 1988.
- [16] Felder, R.M., *Authors preface to "Learning and teaching styles in engineering education"*. June 2002 Available: <http://www4.ncsu.edu/unity/lockers/users/f/felder/public/Papers/LS-1988.pdf>
- [17] Gomez-de-Gabriel, J. M., Mandow, A., Fernandez-Lozano, J., Garcia-Cerezo, A. J. *Using LEGO NXT mobile robots with LabVIEW for undergraduate courses on mechatronics* IEEE Trans. Educ., vol. 54, no. 1, pp. 41–47, Feb. 2011.
- [18] Hong, D. W., Ingram, M., Lahr, D. *Whole skin locomotion inspired by amoeboid motility mechanisms* J. Mech. Robot., vol. 1, no. 1, pp. 011015-1–7, Feb. 2009.
- [19] Imberman, S. P., Klibaner, R. *A robotics lab for CSI*, Journal of Computing Sciences in Colleges, volume 21, number 2, 2005, pp. 131–137, Consortium for Computing Sciences in Colleges, USA.
- [20] Lin, H.-T., Leisk, G. G., Trimmer, B. *GoQBot: a caterpillar-inspired soft-bodied rolling robot* Bioinspir. Biomim., vol. 6, no. 2, pp. 026007-1–14, Jun. 2011.
- [21] Markham, S. A., King, N. K., *Using personal robots in CSI: experiences, outcomes, and attitudinal influences*, Proceedings of the 15th annual conference on Innovation and technology in computer science education, ITiCSE '10, 2010, pp. 204–208, ACM, New York, NY, USA.
- [22] Mayer, R. *Should there be a three-strikes rule against pure discovery learning? The case for guided methods of instruction*. American Psychologist 59 (1): 1419. 2004.
- [23] Murphy, R. R., *Competing for a robotics education*. IEEE Robot. Autom. Mag., vol. 8, no. 2, pp. 44–55, Jun. 2001.
- [24] Nishida, M., Tanaka, K. *The quadruped locomotion robot with the flexible materials* in Proc.IROS, 2008, pp. 2649–2654.
- [25] Nourbakhsh, I., Crowley, K., Bhawe, A., Hamner, E., Hsiu, T., Perez-Bergquist, A., Richards, S., Wilkinson, K. *The Robotic Autonomy Mobile Robotics Course: Robot Design, Curriculum Design and Educational Assessment* In Autonomous Robots, volume 18, number 1, 2005, pp 103–127, Kluwer Academic Publishers, Hingham, MA, USA.
- [26] Petre, M., Price, B. *Using Robotics to Motivate "Back Door" Learning* Education and Information Technologies, volume 9, number 2, pp. 147–158, Kluwer Academic Publishers, Hingham, MA, USA.
- [27] Pfeifer, R., Bongard, J. *How the Body Shapes the Way We Think - A new View of Intelligence* MIT Press (2007) ISBN 978-0-262-16239-5
- [28] Pfeifer, R., Lungarella, M., Iida, F. *Self-organization, embodiment, and biologically inspired robotics* Science 318, 1088–1093
- [29] Pfeifer, R., Gomez, G. *Morphological computation – connecting brain, body, and environment* In B. Sendhoff, O. Sporns, E. Körner, H. Ritter, & K. Doya, K. (eds.), *Creating Brain-like Intelligence: From Basic Principles to Complex Intelligent Systems* (pp.66–83). Berlin: Springer.

<sup>6</sup><http://shanghailectures.org>



- [30] Pekárová, J., *Using a Programmable Toy at Preschool Age: Why and How?*, Workshop Proceedings of SIMPAR 2008 Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots, pp 112–121, 2008, ISBN 978-88-95872-01-8.
- [31] Pinto, M., Moreira, A. P., Matos, A. *Localization of mobile robots using an extended Kalman filter in a LEGO NXT* IEEE Trans. Educ., vol. 55, no. 1, pp. 135-144, Feb. 2012.
- [32] Rawat, K. S., Massiha, G. H. *A hands-on laboratory based approach to undergraduate robotics education* in Proc. ICRA, 2004, pp. 1370-1374.
- [33] Rinderknecht, M., Ruesch, J., Hadorn, M. *The Lagging Legs Exploiting Body Dynamics to Steer a Quadrupedal Agent* In proceedings of the International Conference on Morphological Computation (ICMC), Venice, Italy, March 2007
- [34] Riojas, M., Lysecky, S., Rozenblit, J. *Educational technologies for precollege engineering education* IEEE Trans. Learn. Technol., vol. 5, no. 1, pp. 20-37, Jan.-Mar. 2012.
- [35] Rosenblatt, M., Choset, H. *Designing and implementing hands-on robotics labs* IEEE Intell. Syst., vol. 15, no. 6, pp. 32-39, Nov.-Dec. 2000.
- [36] Seok, S., Onal, C. D., Wood, R., Rus, D., Kim, S. *Peristaltic locomotion with antagonistic actuators in soft robotics* in Proc. ICRA, 2010, pp. 1228-1233.
- [37] Shepherd, R. F., Ilievski, F., Choi, W., Morin, S. A., Stokes, A. A., Mazzeo, A. D., Chen, X., Wang, M., Whitesides, G. M. *Multigait soft robot* Proc. Natl. Acad. Sci. U. S. A., vol. 108, no. 51, pp. 20400-20403, Dec. 2011.
- [38] Steltz, E., Mozeika, A., Rodenberg, N., Brown, E., Jaeger, H. M. *JSEL: Jamming skin enabled locomotion* in Proc.IROS, 2009, pp. 5672-5677.
- [39] Stoeckelmayr, K., Tesar, M., Hofmann, A. *Kindergarten Children Programming Robots: A First Attempt*, pp 185–192, 2011, INNOC – Austrian Society for Innovative Computer Sciences.
- [40] Trivedi, D., Rahn, C. D., Kier, W. M., Walker, I. D. *Soft robotics: Biological inspiration, state of the art, and future research* Applied Bionics and Biomechanics, vol. 5, no. 3, pp. 99-117, Sep. 2008.
- [41] Umaschi Bers, U., *Blocks to Robots: Learning with Technology in the Early Childhood Classroom*, NY: Teachers College Press, 2008.
- [42] Wait, K. W., Jackson, P. J., Smoot, L. S. *Self locomotion of a spherical rolling robot using a novel deformable pneumatic method* in Proc. IRCA, 2010, pp. 3757-3762.



## Appendix E

---

# Curriculum Vitae

## Dorit Assaf

### Address

Ackersteinstrasse 111  
8049 Zurich  
Switzerland  
+41 (0)78 830 55 54  
assaf@ifi.uzh.ch

### Education

- |                                                                                                                                                                                                                                |                                |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------|
| <b>PhD studies</b>                                                                                                                                                                                                             | October 2006 - (December 2012) |
| Artificial Intelligence Laboratory (AILAB)<br>Department of Informatics<br>University of Zurich, Switzerland                                                                                                                   |                                |
| <b>M.S. in Computer Science and Business Informatics</b>                                                                                                                                                                       | June 2005                      |
| Dipl. Inform. Wirtschaftsinformatik<br>Faculty of Economics, Business Administration and Information Technology<br>University of Zurich, Switzerland<br>Concentrations in Artificial Intelligence (AI) and Distributed Systems |                                |
| <b>Matura with major in natural sciences (Typus C)</b>                                                                                                                                                                         | February 1998                  |
| Swiss Secondary School (i.e. qualification for university entrance)<br>Freies Gymnasium Zürich<br>Zurich, Switzerland                                                                                                          |                                |

### Experience

- |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                                 |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------|
| <b>PhD candidate</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | October 2006 - to present       |
| Artificial Intelligence Laboratory (AILAB)<br>Department of Informatics<br>University of Zurich, Switzerland                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                 |
| <ul style="list-style-type: none"> <li>• Development of an open-source, modular robot kit for education.</li> <li>• Design and production of printed circuit boards (embedded systems using Atmel AVR microcontrollers).</li> <li>• Software development (Java application on client, firmware on microcontrollers (AVR C/C++)).</li> <li>• Used communication technologies: CAN, XBee, Bluetooth, USB, RS232.</li> <li>• Rapid prototyping using Solidworks CAD software and 3D printing technologies.</li> <li>• Focus on customization of and contribution to existing open-source projects such as Arduino.</li> <li>• Lecturer AI seminar (creating curriculum, teaching, supervision).</li> <li>• Student supervision (for bachelor and master theses, semester projects).</li> <li>• Teaching assistance (creating and grading exercises and exams of undergraduates).</li> <li>• Research practice (e.g. writing grant proposals, reviews, publications, attending conferences etc.).</li> </ul> |                                 |
| <b>System Test Engineer</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | September 2005 - September 2006 |
| Tecan Schweiz AG<br>Männedorf, Switzerland<br>Company working in the field of laboratory automation                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                 |
| <ul style="list-style-type: none"> <li>• System test engineer within R&amp;D department of the biopharmaceutical branch of the company.</li> <li>• Creation of test cases for new products based on specifications.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                 |

- Planning and conduction of tests including their documentation (writing and running test scripts using corresponding in-house software).
- Setting up the robotic systems (hardware, software, and firmware) for the test runs.
- Finding software bugs and creation of software change requests.
- Close collaboration with software developers and product managers.
- Compliance with company policy and ISO standard.
- 3rd level support.
- Helping building and maintaining the system test infrastructure.

**Internship as Software Developer / Research Assistant** April - July 2004  
 Faculty of Information Management  
 University of Zurich, Switzerland

- Work in the context of the European research project *Mobillearn*.
- Development of a prototype application for collaborative work on a personal digital assistant (PDA).
- Implementation of a client/server architecture, where platform independent PDAs connected to a server application on a PC through the Internet.
- Used technologies: Java for server application and SuperWaba for clients (open-source IDE for PDAs and smart-phones).

**Secretary and marketing assistance** October 1998 - August 1999  
 Dantronic AG  
 Wetzikon, Switzerland  
 Distributor of industrial electronics

- Handling of incoming quote requests and phone calls.
- Creation of the company's website and brochures.
- Assisting in the preparation and attendance of the company's exhibitions.

## Teaching Activities

**Lecturer in robotics and Artificial Intelligence** January 2010 and 2011  
 Master of Advanced Studies *Ergänzungsfach Informatik an Gymnasien*  
 University of Zurich, Switzerland

- Teaching a module within MAS degree program for secondary school teachers.
- Development of a robotic platform (wheeled robot with different sensors) using Arduino.
- Teaching topics in Artificial Intelligence.
- Teaching C/C++ programming using Arduino.

**Lecturer in designing and programming embedded systems** spring term 2011  
 Seminar for master students  
 AILAB, University of Zurich, Switzerland

- Teaching embedded systems design using Altium Designer.
- Teaching programming microcontrollers using AVR C/C++ and Eclipse.

**Lecturer in Artificial Intelligence** February - April 2010  
 Advanced training program for educators  
 Institut für Erziehungswissenschaft  
 University of Bern, Switzerland

- Teaching a module in AI for an advanced training program for educators.
- Practical C/C++ programming assignments using Arduino.

**Lecturer AI Seminar**

spring term 2007, 2008, 2009

Seminar for bachelor and master students  
AILAB, University of Zurich, Switzerland

- Creation of the seminar curriculum.
- Students built robots using the Asuro robot kit.
- Supervision of students' programming projects.

**Robotics Workshop**

October 2007 and 2008

Advanced training program *Robotik fürs Gymnasium* for teachers  
Autonomous Systems Lab, Institute of Robotics and Intelligent Systems  
ETH Zurich, Switzerland

- Designing and conducting a workshop using the LEGO NXT platform with focus on robot morphology.

**Teaching Assistance**

fall term 2006, 2007, 2008

AI classes for bachelor and master students  
AILAB, University of Zurich, Switzerland

- Designing and grading exercises and exams.
- Assistance of lecture preparation.
- Students supervision.

**Tutor Java Programming**

fall term 2004

Assessment year students' course in Java programming  
Department of Informatics  
University of Zurich, Switzerland

- Weekly two hour tutorial session.
- Students supervision.
- Correction of exercises and exams.

**Affiliations**

Member of *NCCR Robotics* (Swiss National Center of Competence in Research).  
Member of *EUCOG* (European Network for the Advancement of Artificial Cognitive Systems, Interaction and Robotics).

**Key Technologies**

Programming: C/C++, Java, AVR C/C++ (microcontrollers)  
Software: Linux, Eclipse, Solidworks, Altium Designer  
Hardware: Digital electronics, Atmel AVR MCU, Arduino

**Languages**

German: native language.  
English: excellent both written and oral.  
Hebrew: very good oral, basic written.  
French: basic both written and oral.

## Publications

D. Assaf, X. Yu, L. Wang, and F. Iida, (2012). "Teaching soft-robotics: A case study in building elastic creatures that locomote." submitted.

D. Assaf, J. C. Larsen and M. Reichardt (2012), "Extending Mechanical Construction Kits to Incorporate Passive and Compliant Elements for Educational Robotics.", *In Proceedings of 3<sup>rd</sup> International Conference on Robotics in Education (RiE 2012)*, pp. 33–40.

D. Assaf and R. Pfeifer (2011). "Robotics as Part of an Informatics Degree Program for Teachers." *In Proceedings of Society for Information Technology & Teacher Education International Conference (SITE 2011)* (pp. 3128–3133). Chesapeake, VA: AACE.

D. Assaf and R. Pfeifer (2011) "EmbedIT – An Open Robotic Kit for Education" *In Research and Education in Robotics – EUROBOT 2011, Communications in Computer and Information Science*, Volume 161, 29–39

M. Hoffmann and D. Assaf (2011), Tutorial on Embodiment, EUCOG website <http://www.eucognition.org/index.php?page=tutorial-on-embodiment>

D. Assaf (2005) "Entwicklung eines Kontrollsystems zur Steuerung eines hupfenden Roboters ueber unterschiedliche Oberflaechen". Master thesis, Artificial Intelligence Laboratory, Department of Informatics, University of Zurich.

